



NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

DISSERTATION

**SPACECRAFT GUIDANCE STRATEGIES FOR
PROXIMITY MANEUVERING AND CLOSE APPROACH
WITH A TUMBLING OBJECT**

by

George A. Boyarko

March 2010

Dissertation Supervisor:

Marcello Romano

Approved for public release; distribution is unlimited

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE March 2010	3. REPORT TYPE AND DATES COVERED Dissertation	
4. TITLE AND SUBTITLE: Spacecraft Guidance Strategies for Proximity Maneuvering and Close Approach with a Tumbling Object			5. FUNDING NUMBERS	
6. AUTHOR George A. Boyarko			8. PERFORMING ORGANIZATION REPORT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A				
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number _____.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE A	
13. ABSTRACT (maximum 200 words) A six degree of freedom, 20-state model of two spacecraft rendezvous is developed, one of which was controlled and the other considered to be passively tumbling. Solutions that minimize a series of performance indices are obtained for the problem of close approach, up to the point of contact, using a direct optimal control method. The solution is then verified as optimal by way of an indirect method based on the Minimum Principle. Next, a trajectory generation method for spacecraft reorientation is developed, based on a quaternion construction of Inverse Dynamics in the Virtual Domain. This new construction enables development of an Inverse Dynamics in the Virtual Domain rapid-trajectory generation method that exploits the concept of decoupling space and time, for the problem of a spacecraft performing a close approach maneuver to a tumbling object. Finally, the advantages of the new method are demonstrated through simulated scenarios that employ two distinct concepts of closed-loop feedback. The benefits seen by Inverse Dynamics in the Virtual Domain methods include the rapid computational time that allows a feasible solution to be generated, potentially onboard a spacecraft and in closed-loop. Although the Inverse Dynamics in the Virtual Domain method cannot match the true optimal solution, it has several advantages. Rapid computational time and the ability to reshape itself when provided updated state variable information reinforce the overall robustness of the method for safe trajectory planning. The novel trajectory generation method developed is tested using Monte Carlo methods to demonstrate its ability to handle realistic situation with varying initial conditions.				
14. SUBJECT TERMS spacecraft proximity operations, Inverse Dynamics in the Virtual Domain, rapid-trajectory generation, spacecraft rendezvous, spacecraft docking,, autonomous assembly, Pontryagin, Minimum Principle, GPOPS, optimal reorientation, optimal rendezvous, quaternion, polynomial.			15. NUMBER OF PAGES 225	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

**SPACECRAFT GUIDANCE STRATEGIES FOR PROXIMITY MANEUVERING
AND CLOSE APPROACH WITH A TUMBLING OBJECT**

George A. Boyarko
Civil Servant, United States Department of Defense
B.S., University of Cincinnati, 2001
M.S., Case Western Reserve University, 2004

Submitted in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY IN ASTRONAUTICAL ENGINEERING

from the

**NAVAL POSTGRADUATE SCHOOL
March 2010**

Author:

George A. Boyarko

Approved by:

Marcello Romano
Assistant Professor of Mechanical
and Aerospace Engineering
Dissertation Supervisor

Brij Agrawal
Distinguished Professor of
Mechanical and Aerospace
Engineering

Oleg Yakimenko
Professor of Mechanical
and Aerospace Engineering

Frank Chavez
Team Lead, Astrodynamics,
Guidance, Navigation and
Control, AFRL/RVES

Xiaoping Yun
Professor of Electrical and
Computer Engineering

Approved by:

Knox Millsaps, Chair, Department of Mechanical and Aerospace
Engineering

Approved by:

Douglas Moses, Vice Provost for Academic Affairs

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

A six degree of freedom, 20-state model of two spacecraft rendezvous is developed, one of which was controlled and the other considered to be passively tumbling. Solutions that minimize a series of performance indices are obtained for the problem of close approach, up to the point of contact, using a direct optimal control method. The solution is then verified as optimal by way of an indirect method based on the Minimum Principle. Next, a trajectory generation method for spacecraft reorientation is developed, based on a quaternion construction of Inverse Dynamics in the Virtual Domain. This new construction enables development of an Inverse Dynamics in the Virtual Domain rapid-trajectory generation method that exploits the concept of decoupling space and time, for the problem of a spacecraft performing a close approach maneuver to a tumbling object. Finally, the advantages of the new method are demonstrated through simulated scenarios that employ two distinct concepts of closed-loop feedback. The benefits seen by Inverse Dynamics in the Virtual Domain methods include the rapid computational time that allows a feasible solution to be generated, potentially onboard a spacecraft and in closed-loop. Although the Inverse Dynamics in the Virtual Domain method cannot match the true optimal solution, it has several advantages. Rapid computational time and the ability to reshape itself when provided updated state variable information reinforce the overall robustness of the method for safe trajectory planning. The novel trajectory generation method developed is tested using Monte Carlo methods to demonstrate its ability to handle realistic situations with varying initial conditions.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	OPTIMAL CONTROL PROBLEM FORMULATION FOR CLOSE RENDEZVOUS AND DOCKING.....	7
B.	INVERSE DYNAMICS IN THE VIRTUAL DOMAIN	7
1.	The Concept of Inverse Dynamics in the Virtual Domain	7
2.	Attitude Trajectory Generation.....	8
3.	Translational Trajectory Generation.....	8
C.	SCOPE OF THE DISSERTATION	9
II.	FORMULATION AND ANALYSIS OF MATCHING POINTS OF INTEREST IN TWO-SPACECRAFT FOR OPTIMAL RENDEZVOUS.....	11
A.	TWO-SPACECRAFT RENDEZVOUS MODELING AND OPTIMIZATION PROBLEM FORMULATION.....	11
B.	SYNTHESIS OF THE OPTIMAL CONTROL USING MINIMUM PRINCIPLE.....	18
1.	Formulation of the Optimal Control Problem	18
2.	Accounting for a Possible Path Constraint.....	24
3.	Possibility of a Singular Control for a Minimum Time Problem..	25
C.	METHODOLOGY FOR OBTAINING A SOLUTION AND CHECKING THE OPTIMALITY.....	27
1.	Defining a Rendezvous Scenario.....	28
2.	Solving the Problem with the Gauss Pseudospectral Optimization Solver (GPOPS)	28
3.	Verification of the GPOPS Solution with Minimum Principle.....	29
D.	OBTAINING AND ANALYZING MINIMUM QUADRATIC-CONTROL (MINIMUM ENERGY) SOLUTION	31
1.	Minimum Energy Solution with GPOPS.....	31
2.	Verification of the GPOPS Solution with Minimum Principle.....	37
E.	OBTAINING AND ANALYZING THE MINIMUM TIME SOLUTION	41
1.	Minimum Time Solution with GPOPS	41
2.	Verification of the GPOPS Solution with Minimum Principle.....	46
3.	Propagation of the GPOPS Solution	52
III.	OPTIMIZATION OF A SPACECRAFT MANEUVER FOR CLOSE APPROACH AND DOCKING WITH A TUMBLING OBJECT	55
A.	RENDEZVOUS MODELING AND OPTIMIZATION PROBLEM FORMULATION.....	55
B.	MINIMUM PRINCIPLE FORMULATION	57
C.	SOLVING THE PROBLEM NUMERICALLY USING A PSEUDOSPECTRAL METHOD.....	66
1.	Minimum Time.....	66
2.	Minimum Quadratic Control (Minimum Energy)	72

3.	Minimum Absolute Control (Minimum Fuel).....	78
4.	Solution Comparisons and Propagation	85
a.	<i>Solution and Comparison</i>	85
b.	<i>Propagating the Solution</i>	86
IV.	TIME-OPTIMAL REORIENTATION OF A SPACECRAFT USING A DIRECT OPTIMIZATION METHOD BASED ON QUATERNION REPRESENTATION OF THE INVERSE DYNAMICS.....	89
A.	SPACECRAFT MODEL AND ATTITUDE TRAJECTORY OPTIMIZATION PROBLEM	89
B.	SOLVING THE PROBLEM USING THE GAUSS PSEUDOSPECTRAL METHOD.....	91
C.	APPLICATION OF INVERSE DYNAMICS IN THE VIRTUAL DOMAIN APPROACH WITH QUATERNION ATTITUDE REPRESENTATION	98
1.	Quaternion Parameterization	98
2.	Mapping from the Virtual Arc to the Time Domain	103
3.	Inverting the Dynamics	104
4.	Matching Endpoint Conditions	105
5.	Increasing the Polynomial Order	106
D.	SOLVING THE PROBLEM USING IDVD METHOD	108
1.	IDVD Solutions Varying 2nd Derivative Results	109
2.	IDVD Solutions Varying 2nd–3rd Derivative Results.....	116
E.	RESULTS AND COMPARISONS.....	120
V.	RAPID ATTITUDE AND TRANSLATIONAL TRAJECTORY GENERATION	129
A.	INVERSE DYNAMICS IN THE VIRTUAL DOMAIN FORMULATION.....	129
1.	Mapping the Virtual Arc and Specifying Endpoint Values.....	130
2.	Implementation of Quaternion Based Attitude Representation..	132
3.	Alternative Euler Angle Formulation for Attitude Representation.....	132
B.	SOLVING THE IDVD PROBLEM	133
C.	COMPUTATIONAL RESULTS	134
1.	Minimum Energy Cost Varying 3rd Parameter Set Results	134
2.	Minimum Time Cost Varying 3rd Parameter Set Results.....	138
3.	Minimum Fuel Cost Varying 3rd Parameter Set Results	141
4.	Minimum Energy Varying 3rd–4th Parameter Set Results	145
5.	Minimum Time Varying 3rd–4th Parameter Set Results	146
6.	Minimum Fuel Varying 3rd–4th Parameter Set Results	148
7.	Propagated IDVD Solutions.....	149
8.	Summary of Close Approach Results for Baseline Maneuver.....	152
VI.	FURTHER SIMULATIONS AND ANALYSIS.....	155
A.	COMPARISON WITH PREVIOUS STUDIES.....	155
1.	Problem Formulation in Two Dimensions.....	155

2.	Direct Method (GPOPS) Formulation and Results	156
3.	IDVD Formulation and Result	160
4.	Comparisons of Trajectory Generation Methods for 2D Example	162
B.	CYCLICAL NATURE OF PROBLEM SOLUTION PERFORMANCE INDEX	163
C.	FLEXIBILITY AND ROBUSTNESS OF IDVD TECHNIQUE	165
1.	Real Time Trajectory Reshaping	165
a.	<i>Solution Results with Inertia Uncertainty</i>	167
b.	<i>Solution Results with Unknown Constant Torque</i>	171
2.	Resolving and Reoptimizing the Problem in Real Time.....	173
3.	Specific Waypoint Spacing in the Time Domain.....	175
D.	EXTENDED EXAMPLES USING INVERSE DYNAMICS IN THE VIRTUAL DOMAIN	178
VII.	CONCLUSION	185
A.	SUMMARY OF RESEARCH	185
B.	SUMMARY OF CONTRIBUTIONS.....	185
C.	CONSIDERATIONS AND RECOMMENDATIONS.....	189
D.	POSSIBLE FUTURE DEVELOPMENTS	192
	LIST OF REFERENCES	193
	APPENDIX. QUATERNION PROPERTIES.....	199
	INITIAL DISTRIBUTION LIST	201

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

Figure 1.	Breakdown of rendezvous and docking mission elements (After Fehse 2003).	4
Figure 2.	Depiction of the spacecraft rendezvous problem.....	12
Figure 3.	The desired final state of two spacecraft system.	12
Figure 4.	Simulink model for integrating the states and costates using the optimal switching function for controls.	29
Figure 5.	Minimum energy solution (GPOPS): The 3D view of the optimal trajectory. A close-up of the final position is shown in the exploded view....	32
Figure 6.	Minimum energy solution (GPOPS): the planar views of the optimal trajectory.	33
Figure 7.	Minimum energy solution (GPOPS): control and associated costate history.	34
Figure 8.	Minimum energy solution (GPOPS): time history of discrepancies in the position and velocity of the chase and RSO docking points.....	34
Figure 9.	Minimum energy solution (GPOPS): history of the transversality conditions.....	35
Figure 10.	Minimum energy solution (GPOPS): history of the Hamiltonian.	36
Figure 11.	Minimum energy solution (MP): Control time histories and associated costates.....	38
Figure 12.	Minimum energy solution (MP): state and costate time-histories for translational variables of the chaser vehicle.	38
Figure 13.	Minimum energy solution (MP): state and costate histories for the defining angular parameters of the chaser vehicle.....	39
Figure 14.	Minimum energy solution (MP): state and costate time histories for the RSO.....	39
Figure 15.	Hamiltonian for the forward shooting minimum-control solution.	40
Figure 16.	Minimum time solution (GPOPS): The 3D view of the optimal trajectory. A close-up of the final position is shown in the exploded view.	42
Figure 17.	Minimum time solution (GPOPS): the 2D plots of optimal rendezvous trajectory.	43
Figure 18.	Minimum time solution (GPOPS): the control and associated costate history.	43
Figure 19.	Minimum time solution (GPOPS): time history of discrepancies in the position and velocity of the chaser and RSO docking point.....	44
Figure 20.	Minimum time solution (GPOPS): history of transversality conditions.....	44
Figure 21.	Minimum time solution (GPOPS): history of the Hamiltonian.....	45
Figure 22.	Minimum time solution (GPOPS): The associated costate history resulting from a 25 node solution.	46
Figure 23.	Minimum time solution (GPOPS): Hamiltonian for the associated 25-node solution.....	46
Figure 24.	Minimum time solution (MP): 3D optimal rendezvous trajectory.	48

Figure 25.	Minimum time solution (MP): The 2D projections of the optimal rendezvous trajectory.	49
Figure 26.	Minimum time solution (MP): control histories and associated switching conditions.	49
Figure 27.	Minimum time solution (MP): State and costate histories for the translational variables of the chaser vehicle.	50
Figure 28.	Minimum time solution (MP): state and costate histories for the defining angular parameters of the chaser vehicle.	50
Figure 29.	Minimum time solution (MP): state and costate time histories for the RSO...	51
Figure 30.	Minimum time solution (MP): Hamiltonian for the forward shooting minimum time solution.	52
Figure 31.	Propagated trajectory using the minimum time control history from GPOPS.	53
Figure 32.	Propagated trajectory using the minimum time control history from GPOPS.	54
Figure 33.	Minimum time solution (GPOPS): 3D plot of optimal rendezvous trajectory.	67
Figure 34.	Minimum time solution (GPOPS): 2D plots of optimal rendezvous trajectory.	67
Figure 35.	Minimum time solution (GPOPS): control history solution.	68
Figure 36.	Minimum time solution (GPOPS): state and costate histories for the defining translational variables of the chaser vehicle.	68
Figure 37.	Minimum time solution (GPOPS): state and costate histories for the defining angular parameters of the chaser vehicle.	69
Figure 38.	Minimum time solution (GPOPS): state and costate histories for the defining angular parameters of the RSO.	69
Figure 39.	Minimum time solution (GPOPS): time history of the translational endpoint discrepancies.	70
Figure 40.	Minimum time solution (GPOPS): time history of the attitude endpoint discrepancies.	70
Figure 41.	Minimum time solution (GPOPS): history of transversality conditions.	71
Figure 42.	Minimum time solution (GPOPS): history of the Hamiltonian.	71
Figure 43.	Minimum energy solution (GPOPS): 3D plot of optimal trajectory.	73
Figure 44.	Minimum energy solution (GPOPS): 2D plots of optimal trajectory.	73
Figure 45.	Minimum energy solution (GPOPS): control history with respect to the optimal trajectory.	74
Figure 46.	Minimum energy solution (GPOPS): state and costate histories for the translational variables of the chaser vehicle.	74
Figure 47.	Minimum energy solution (GPOPS): state and costate histories for the defining angular parameters of the chaser vehicle.	75
Figure 48.	Minimum energy solution (GPOPS): state and costate histories for the defining angular parameters of the RSO.	75
Figure 49.	Minimum energy solution (GPOPS): history of the translational endpoint discrepancies.	76

Figure 50.	Minimum energy solution (GPOPS): history of the rotational endpoint discrepancies.....	76
Figure 51.	Minimum energy solution (GPOPS): history of the transversality conditions.....	77
Figure 52.	Minimum energy solution (GPOPS): history of the Hamiltonian.....	77
Figure 53.	Minimum fuel solution (GPOPS): 3D plot of optimal rendezvous trajectory.....	80
Figure 54.	Minimum fuel solution (GPOPS): 2D plots of optimal rendezvous trajectory.....	80
Figure 55.	Minimum fuel solution (GPOPS): control history with respect to the optimal trajectory.....	81
Figure 56.	Minimum fuel solution (GPOPS): state and costate histories for the translational variables of the chaser vehicle.....	81
Figure 57.	Minimum fuel solution (GPOPS): state and costate histories for the defining angular parameters of the chaser vehicle.....	82
Figure 58.	Minimum fuel solution (GPOPS): state and costate histories for the RSO....	82
Figure 59.	Minimum fuel solution (GPOPS): history of the translational endpoint discrepancies.....	83
Figure 60.	Minimum fuel solution (GPOPS): history of the rotational endpoint discrepancies.....	83
Figure 61.	Minimum fuel solution (GPOPS): history of the transversality conditions....	84
Figure 62.	Minimum fuel solution (GPOPS): history of the Hamiltonian.....	84
Figure 63.	Propagated trajectory using the minimum time control history from GPOPS.....	87
Figure 64.	Propagated trajectory using the minimum time control history.....	87
Figure 65.	Endpoint conditions on chaser and RSO docking position, velocity, angular position and angular rate.....	88
Figure 66.	Case 1 (GPOPS solution): time histories of the state variables.....	93
Figure 67.	Case 1 (GPOPS solution): time history of the controlling torques.....	93
Figure 68.	Case 1 (GPOPS solution): the 3D representation of the solution.....	94
Figure 69.	Case 1 (GPOPS solution): comparison of time histories of angular velocity components obtained for 25 nodes (top) and 50 nodes (bottom).....	95
Figure 70.	Case 1 (GPOPS solution): comparison of time histories of torques, obtained for 25 nodes (top) and 50 nodes (bottom).....	96
Figure 71.	Case 1 (GPOPS solution): the 3D representation of the 25-node solution....	96
Figure 72.	Case 2 (GPOPS solution): time histories of the state variables.....	97
Figure 73.	Case 2 (GPOPS solution): time history of the controlling torques.....	97
Figure 74.	Value of 5th order Bezier Polynomials with respect to the virtual arc.....	100
Figure 75.	Value of 7th order Bezier Polynomials with respect to the virtual arc.....	107
Figure 76.	Case 1 (IDVD 5th order): time history of the quaternion.....	109
Figure 77.	Case 1 (IDVD 5th order): time histories of the angular velocity and accelerations.....	110
Figure 78.	Case 1 (IDVD 5th order): time history of controlling torques.....	110
Figure 79.	Case 1 (IDVD 5th order): mapping the virtual and time.....	111
Figure 80.	Case 1 (IDVD 5th order): principal axis outline of 180° slew.....	111

Figure 81.	Case 1 (IDVD 5th order): time history of the quaternion using 25 nodes.....	112
Figure 82.	Case 1 (IDVD 5th order): time histories of the angular velocity and acceleration using 25 nodes.	112
Figure 83.	Case 1 (IDVD 5th order): time history of controlling torques using 25 nodes.	112
Figure 84.	Case 1 (IDVD 5th order): mapping the virtual and time domains using 25 nodes.	113
Figure 85.	Case 1 (IDVD 5th order): comparison of angular velocity time histories, obtained for 25 nodes (top) and 50 nodes (bottom).	114
Figure 86.	Case 1 (IDVD 5th order): comparison of torque time histories, obtained for 25 nodes (top) and 50 nodes (middle).	114
Figure 87.	Case 2 (IDVD 5th order): Time history of the quaternion.....	115
Figure 88.	Case 2 (IDVD 5th order): time histories of the angular velocity and acceleration.	115
Figure 89.	Case 2 (IDVD 5th order): time history of controlling torques.....	116
Figure 90.	Case 2 (IDVD 5th order): mapping the virtual and time domains.....	116
Figure 91.	Case 1 and 2 (IDVD 7th order): time history of the quaternions Case 1 (top) and Case 2 (bottom).	117
Figure 92.	Case 1 and 2 (IDVD 7th order): time histories of the angular velocity and acceleration for Case 1 (top) and Case 2 (bottom).	118
Figure 93.	Case 1 and 2 (IDVD 7th order): time history of controlling torques for Case 1 (top) and Case 2 (bottom).	119
Figure 94.	Case 1 and 2 (IDVD 7th order): mapping the virtual and time domains for Case 1 (top) and Case 2 (bottom) solutions.	120
Figure 95.	Case 1: 180° rest-to-rest slew profile as projected onto the x-y inertial plane.....	126
Figure 96.	Case 1: 90° rest-to-rest slew profile as projected onto the x-y inertial plane.....	127
Figure 97.	Case 1: 90° slew profile, $\omega_0 = -\omega_f = [0.1 \ 0.1 \ 0.1]^T$, as projected onto the x-y inertial plane.	128
Figure 98.	Basis functions considered for translational trajectory generation.	130
Figure 99.	Minimum energy solution (IDVD 3rd): 3D optimal rendezvous trajectory. .	135
Figure 100.	Minimum energy solution (IDVD 3rd): 2D planar projection of the trajectory.	135
Figure 101.	Minimum energy solution (IDVD 3rd): control history.	136
Figure 102.	Minimum energy solution (IDVD 3rd): history of the translational endpoint discrepancies.	136
Figure 103.	Minimum energy solution (IDVD 3rd): time history of the rotational endpoint discrepancies.	137
Figure 104.	Minimum energy solution (IDVD 3rd): time history of the speed factor.....	137
Figure 105.	Minimum time solution (IDVD 3rd): the 3D optimal rendezvous trajectory.	138
Figure 106.	Minimum time solution (IDVD 3rd): 2D planar projection of the trajectory.	139
Figure 107.	Minimum time solution (IDVD 3rd): control history.	139

Figure 108.	Minimum time solution (IDVD 3rd): history of the translational endpoint discrepancies.	140
Figure 109.	Minimum time solution (IDVD 3rd): history of the rotational endpoint discrepancies.	140
Figure 110.	Minimum time solution (IDVD 3rd): time history of the speed factor.	141
Figure 111.	Minimum fuel solution (IDVD 3rd): the 3D optimal rendezvous trajectory.	142
Figure 112.	Minimum fuel solution (IDVD 3rd): 2D planar projection of the trajectory.	142
Figure 113.	Minimum fuel solution (IDVD 3rd): control history.	143
Figure 114.	Minimum fuel solution (IDVD 3rd): history of the translational endpoint discrepancies.	143
Figure 115.	Minimum fuel solution (IDVD 3rd): history of the rotational endpoint discrepancies.	144
Figure 116.	Minimum time solution (IDVD 3rd): time history of the speed factor.	144
Figure 117.	Minimum energy solution (IDVD 3rd–4th): 3D optimal rendezvous trajectory.	145
Figure 118.	Minimum energy solution (IDVD 3rd–4th): control history.	146
Figure 119.	Minimum time solution (IDVD 3rd–4th): 3D optimal rendezvous trajectory.	147
Figure 120.	Minimum time solution (IDVD 3rd–4th): control history.	147
Figure 121.	Minimum fuel solution (IDVD 3rd–4th): 3D optimal rendezvous trajectory.	148
Figure 122.	Minimum fuel solution (IDVD 3rd–4th): control history.	149
Figure 123.	Translational discrepancies for propagated IDVD (3rd) minimum time solution.	150
Figure 124.	Rotational discrepancies for propagated IDVD (3rd) minimum time solution.	150
Figure 125.	Control history of different methods for the minimum quadratic-control problem formulation.	153
Figure 126.	Control history of different methods for the minimum time problem formulation.	153
Figure 127.	Control history of different methods for the minimum fuel problem formulation.	154
Figure 128.	2D Case (GPOPS): optimal minimum time trajectory of the chaser in the RSO body frame.	157
Figure 129.	2D Case (GPOPS): optimal minimum time control history.	157
Figure 130.	2D Case (GPOPS): history of the translational endpoint conditions.	158
Figure 131.	2D Case (GPOPS): history of the attitude endpoint conditions.	158
Figure 132.	2D Case (GPOPS): history of the transversality conditions.	159
Figure 133.	2D Case (GPOPS): history of the Hamiltonian.	159
Figure 134.	2D Case (IDVD): minimum time trajectory of the chaser in the RSO body frame.	160
Figure 135.	2D Case (IDVD 3rd): minimum time control history.	161
Figure 136.	2D Case (IDVD 3rd): history of the translational endpoint conditions.	161
Figure 137.	2D Case (IDVD 3rd): history of the attitude endpoint conditions.	162

Figure 138.	Energy and fuel costs for rendezvous with a tumbling RSO with symmetric inertia matrix. The time shown is a wait time until the start of the maneuver.....	164
Figure 139.	Energy and fuel costs for rendezvous with a tumbling RSO with an asymmetric inertia matrix. The time shown is the total maneuver time.....	164
Figure 140.	Block diagram of real time trajectory reshaping concepts using IDVD.....	167
Figure 141.	Results of the final trajectory for the close approach example having a misidentified inertia matrix IDVD real time reshaping.	169
Figure 142.	Exploded view of the RSO docking point information for a misidentified inertia matrix example.	170
Figure 143.	Spatial views of the resulting trajectories for the close approach example having a misidentified inertia matrix IDVD real time reshaping.	171
Figure 144.	View of the close approach trajectory example with an unidentified, constant torque using IDVD real time reshaping.....	172
Figure 145.	Spatial views of the resulting trajectories for the close approach example having an unidentified constant torque using IDVD real time reshaping.....	173
Figure 146.	Block diagram of real time trajectory recalculation concepts using IDVD...	174
Figure 147.	(Top) Overlaid recomputed trajectories. (Bottom) Resulting overall trajectory based on the most current recalculation.	175
Figure 148.	Control history of a representative trajectory with uniform node spacing in the virtual domain.	176
Figure 149.	Control history of a representative trajectory with uniform node spacing in the time domain.....	177
Figure 150.	Trajectories for sample rendezvous problem for the first 10 random starting chaser positions with zero initial relative velocity.....	179
Figure 151.	Trajectories for sample rendezvous problem for the first 10 random starting chaser positions with nonzero initial relative velocity.....	180
Figure 152.	Distribution of the 1000 pseudorandom initial conditions tested.	181
Figure 153.	Planar view of the 1000 pseudorandom initial conditions tested.	181
Figure 154.	Distribution of the initial chaser range for the 1000 pseudorandom samples generated.	182
Figure 155.	Distribution of the initial chaser velocity for the 1000 pseudorandom samples generated.	182
Figure 156.	Distribution of the initial RSO angular velocity for the 1000 pseudorandom samples generated.....	184

LIST OF TABLES

Table 1.	The initial values of the states.....	28
Table 2.	The initial values of costates and final time as defined by GPOPS.....	35
Table 3.	Value of terminal and transversality conditions at the final time.	35
Table 4.	The initial values of costates and t_f as defined by MP.	40
Table 5.	The value of terminal and transversality conditions at the final time.....	40
Table 6.	The initial values of costates and as defined by GPOPS.	45
Table 7.	Value of terminal conditions at the final time as calculated by GPOPS.	45
Table 8.	The initial values of costates and t_f as defined by MP.	51
Table 9.	Value of terminal and transversality conditions at the final time.	51
Table 10.	Value of terminal conditions at the final time.	54
Table 11.	The initial values of costates and t_f as defined by GPOPS for minimum time solution.....	72
Table 12.	The initial values of costates and t_f as defined by GPOPS for minimum energy (quadratic-control) solution.....	78
Table 13.	The initial values of costates and t_f as defined by GPOPS for minimum fuel solution.	85
Table 14.	Comparison of solutions for the three optimal control problems.	85
Table 15.	Value of terminal conditions at the final time.	88
Table 16.	Description of the test cases.....	91
Table 17.	The 180° rest-to-rest slew maneuver about the z-axis, symmetric inertia (Case 1).	121
Table 18.	The 180° rest-to-rest slew maneuver about the z-axis, asymmetric inertia (Case 2).	122
Table 19.	The 90° rest-to-rest slew maneuver about the z-axis, symmetric inertia (Case 1).	123
Table 20.	The 90° rest-to-rest slew maneuver about the z-axis, asymmetric inertia (Case 2).	124
Table 21.	The 90° maneuver for symmetric inertia (Case 1) and nonzero boundary rates, $\omega_0 = -\omega_f = [0.1 \ 0.1 \ 0.1]^T$	124
Table 22.	The 90° maneuver for symmetric inertia (Case 1) and nonzero boundary rates, $\omega_0 = -\omega_f = -[1 \ 1 \ 1]^T$	125
Table 23.	Value of terminal conditions at the final time for IDVD (3rd) minimum time solution.....	151
Table 24.	Value of terminal conditions at the final time for IDVD (3rd) minimum energy solution.....	151
Table 25.	Value of terminal conditions at the final time for IDVD (3rd) minimum fuel solution.	151
Table 26.	Summary of performance indices and computational time for a variety of trajectory generation methods.....	152

Table 27.	Summary of performance indices and computational time for a 2D scenario. *A factor of 0.005 was multiplied by the minimum fuel cost to be consistent with the formulation presented by Ma (2007).....	163
Table 28.	Summary of performance indices and computational time for IDVD (3rd) and IDVD (3rd–4th) method using 1000 pseudorandom samples for varying the initial conditions of the chaser position and velocity.	183
Table 29.	Summary of performance indices and computational time for IDVD (3rd) and IDVD (3rd–4th) method using 1000 pseudorandom samples for varying the initial conditions of the chaser position but having the initial relative velocity be equal to zero.	183
Table 30.	Summary of performance indices and computational time for IDVD (3rd) and IDVD (3rd–4th) method using 1000 pseudorandom samples for varying the initial conditions for position and velocity of the chaser and initial angular velocity of the RSO.	184

LIST OF ACRONYMS AND ABBREVIATIONS

2D	Two-dimensional
3D	Three-dimensional
ADCS	Attitude Determination Control Subsystem
AFRL	Air Force Research Laboratory
ATV	Automated Transfer Vehicle
C	Chaser Spacecraft
CM	Center of Mass
CONOPS	Concept of Operations
CPU	Central Processing Unit
DARPA	Defense Advanced Research Projects Agency
DART	Demonstration for Autonomous Rendezvous Technology
DCM	Direction Cosine Matrix
DoD	Department of Defense
DoF	Degree of Freedom
GNC	Guidance, Navigation & Control
GPOPS	Gauss Pseudospectral Optimization Software
IDVD	Inverse Dynamics in the Virtual Domain
LEO	Low Earth Orbit
LQG	Linear Quadratic Gaussian
LQR	Linear Quadratic Regulator
MED	Momentum Exchange Devices
MP	Minimum Principle
NASA	National Aeronautics and Space Administration
NPS	Naval Postgraduate School
NRL	Naval Research Laboratory
ODE3	(MATLAB) Ordinary Differential Equation Solver 3
OE	Orbital Express

PI	Performance Index
RSO	Resident Space Object
RW	Reaction Wheel
SNOPT	Sparse NOlinear OPTimizer
SPHERES	Synchronized Position Hold Engage and Reorient Experimental Satellites
SRL	Spacecraft Robotics Laboratory
SS	Sakawa Shindo
SUMO	Spacecraft for the Universal Modification of Orbits
T	Target Spacecraft
TacSat	Tactical Satellite
TPBVP	Two-Point Boundary-Value Problem
XSS	Experimental Satellite System

ACKNOWLEDGMENTS

I would like to acknowledge the United States Air Force Research Laboratory for giving me the opportunity to attend the United States Naval Postgraduate School. Specifically, I would like to thank Dr. Richard de Jonckheere for his efforts in starting the process, Mr. Kevin Schum for seeing the process through to completion, and Dr. Jim Riker and Col. Wade Vaught for realizing the importance of the unique programs at NPS.

I would like to thank my dissertation committee and instructors for their unwavering support. Specifically, I thank Prof. Marcello Romano for taking me on as a student and providing invaluable guidance as my advisor; Prof. Oleg Yakimenko for the countless hours of personal time he spent working with me and his drive to challenge me to the fullest; Prof. Xiaoping Yun for all the attention and help he has provided as well as training me to keep an open mind; Dist. Prof. Brij Agrawal for accepting me into the program and helping to make the overall goal a reality; Dr. Frank Chavez for reminding me to look at the big picture and keeping me grounded; Prof. Ravi Vaidyanathan for his personal interest in my success; and Lt. Col. Tim Sands, Lt. Col. Shawn McCamish, LCDR Jason Hall, Maj. Dan Burtz, and Capt. Melissa Corley for their help along me the way.

I want to thank my colleagues in the Astronautical Engineering and 591 curriculum class of 2007 and their families, for accepting me into their social network with their home-cooked meals, crawfish boils, surprise birthday parties, college football events, bonfires, softball games, and card nights that helped me stay sane throughout the entire residency process away from my family.

I want to thank my children, Drew and Pam, for their positive attitudes and welcoming of dad home after many trips to California. I want to thank my parents and brother for their invaluable help and encouragement.

Finally, I would like to thank my beautiful, loving wife, Colleen, for her support. She is the only person who can truly appreciate all of the derivations, formulations, and efforts that did not work (as well as the accompanying frustrations and caffeine binges) before arriving at something that did. Without her, this would not have been possible.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

The rendezvous problem of two spacecraft orbiting the earth has been addressed in numerous publications, research thrusts and desired mission capabilities. Rendezvous technology has also evolved with small spacecraft development, such as the National Aeronautics and Space Administration's (NASA) DART (NASA 2004), Air Force Research Laboratory's (AFRL) (Experimental Satellite System) XSS-10 (Davis 2003) and XSS-11 (AFRL 2005), Naval Research Laboratory (NRL) SUMO (Bosse 2004), and Defense Advanced Research Projects Agency's (DARPA) Orbital Express (OE) (Kennedy 2008). In particular, the AFRL Space Vehicle Directorate at Kirtland Air Force Base in New Mexico developed the XSS-11 in order to exhibit the ability for a small satellite to autonomously plan and rendezvous with a passive and cooperative Resident Space Object (RSO) in low Earth orbit (LEO). XSS-10 was a simple proximity mission around the upper stage that it was boosted into orbit on. XSS-11 demonstrated the ability to change orbits and intercept other nonrelated objects. The use of micro-satellites to inspect, service, repair, and refuel larger spacecraft is a long-term goal. The closest the XSS-11 approached and maneuvered around another object in space was approximately 500 meters. In addition, DARPA's OE Advanced Technology Demonstration Program validated the technology and techniques for on-orbit refueling and reconfiguration of two satellites. The mission, conducted in 2007, performed several autonomous rendezvous and capture scenarios, including component exchange and propellant transfer events. The existence of these programs demonstrates that there is a need for a robust and effective autonomous close proximity control algorithm for multiple small spacecraft. Still, even with the investment in the aforementioned missions, spacecraft proximity operations that have any dynamic tasking attributes are currently executed with humans in the loop, relying heavily on an extensive operations staff to plan and oversee the maneuvers, and performing maneuvers while in communication with ground stations. This approach is extremely cumbersome with respect to time, manpower and overall operational footprint.

From a historical point of view, one of the first attempts to recapture a satellite in orbit was in 1992 with Intelsat6 from the Space Shuttle (Broad 1992). In this case, jet thrusters on both the shuttle and the satellite were repeatedly fired during the three-day operation to bring the two spacecraft to a complex orbital rendezvous. The ground controllers of Intelsat had slowed the rotation of satellite to 0.67 revolutions per minute so it would be easier to grasp. Still, astronauts in spacesuits tried and failed twice capture the satellite from orbit and move it into the payload bay of the space shuttle Endeavour. Success was eventually realized on the third try.

The vast majority of previous research pertaining to operations that focused on rendezvous with an uncontrolled rotating object has only addressed the translational challenges (Jacobsen, Lee, Zhu and Dubowsky 2002; Matsumoto, Dubowsky, Jacobsen, and Ohkami 2003a and 2003b). This also involved segmented maneuvers and only matching the translational position and velocity of a point mass with that of a docking point on a tumbling RSO. Nolet, Kong, and Miller used a simple glideslope algorithm for the purpose of attempting to generate trajectories for the two-dimensional (2D) case with hopes of implementation on the Massachusetts Institute of Technology SPHERES testbed (2004). Huntington (Huntington and Rao 2008b) and Singh (Singh and Hadaegh 2001) independently researched proximity operations but with cooperative agents. Henshaw mentions optimal rendezvous with a tumbling object, but there is no formal presentation of the results using the Minimum Principle (MP) (Henshaw 2003). Henshaw's method is a variation trajectory planner capable of planning large timescale maneuvers (on the order of 10,000+ seconds to several years), therefore there is no discussion of the computational times involved to obtain a solution.

The optimal satellite reorientation problem alone is of general interest to many in the field of aerospace engineering. Its interest increases as we start to consider autonomous rendezvous and close approach mission scenarios that couple both attitude and translational dynamics with the desire to match both translational and angular motion in preparation for docking. The available literature on optimal and efficient spacecraft reorientation alone is extensive (Junkins and Turner 1986; Bilimoria and Wie 1993; Vadali and Junkins 1984). Many civilian and military space missions need to have agile

attitude maneuver capability. For instance, Tactical Satellite (TacSat) 3 was intended to demonstrate responsive delivery of information to operational users (Davis and Straight 2006). Due to the satellite's Low Earth Orbit, the timeline for tasking, slewing and disseminating data is greatly reduced. Other challenges include the fact that the tasking can be modified at any moment until after a short period of time with the ground station who is uploading the tasking and the idea that TacSat must autonomously slew to the target, collect and process the data and then down link the data directly to the customer who is not collocated with the ground station. Current real time feedback controls are not optimized for minimum time (Vadali and Junkins 1984). These challenges lend themselves to the need for the ability to rapidly generate feasible trajectories that are optimized for minimum time.

The preceding examples bring about a very interesting concept: what if only a feasible solution is desired, with little or no regard for maneuver time (aside from a maximum final time) or power used, where mission success is the only criterion evaluated? Is the new metric for optimization based on computational time or robustness? Such is the case for the reorientation of a Tactical Satellite (TacSat) (Davis and Straight 2006). The desire is to perform the mission within certain operating constraints. If the satellite happens to reorient in minimum time, nothing will be gained (as long as all other constraints on the dynamics are met) because the target or the downlink may not be in view, or the elevation mask to the target may not be cleared. For certain missions, due to target spacing, access times and general Concept of Operations (CONOPS), a hurry-up and wait mentality will not benefit the mission as much as rapidly generating the trajectory for the satellite to be dynamically tasked at the latest possible instant (not orbits ahead of time). In order for the satellite to perform the maneuver, with potentially target locations dynamically changing up until the point the link to the groundstation is broken, a feasible trajectory needs to be calculated in the minimum amount of time possible before the first target acquisition. In essence, there is no time to wait for a detailed optimal solution to be computed, but the implementation of the best optimized solution at that time is of utmost importance. This concept will be revisited throughout the different subsections of this dissertation where the rapid generation of the

solution and robustness of always having a feasible trajectory to follow, is considered more important than the performance index (PI), such as minimum time, minimum energy, etc., to be optimized.

The entire mission of rendezvous and docking can be broken down into several components and subcomponents. Figure 1, modified from Fehse (2003), illustrates the various segments of the mission.

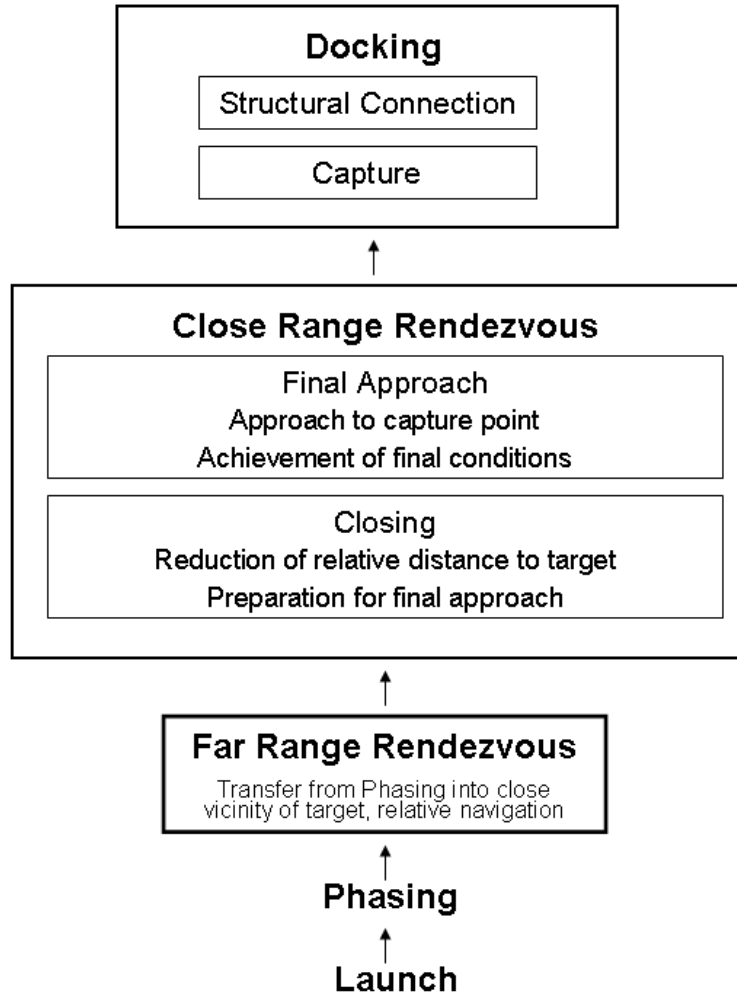


Figure 1. Breakdown of rendezvous and docking mission elements (After Fehse 2003).

The work in this dissertation revolves around the desire to efficiently perform close range rendezvous maneuvers to a tumbling spacecraft for the purpose of close inspection or docking. The problem of docking, contact and grappling once the chaser

satellite is in position is currently being explored by NRL (Creamer 2007; Tasker and Henshaw 2008). Among others, research by McCamish (2007) addresses the problem of efficiently and safely taking a satellite from far range into close proximity, or closing, to an RSO, albeit a nontumbling one. Therefore, this research fits in the realm of close approach trajectory generation, or more specifically final approach, for the purpose of close inspection or docking with a tumbling RSO. Launch and Phasing (maneuvering to match orbital elements and far approach) are assumed to have already taken place. Specifically, adapting the terminology set forth by Nolet (2007), this is subset of the terminal phase, starting ~10m from the RSO up until docking contact has been initiated. In concert with previous research in this area, it is not the intent to advance navigation techniques for determining the attributes of a tumbling RSO, therefore it assumes that exact knowledge of the RSO states are available. Complementary research efforts are looking at the problem of determining the states of an RSO (Jasiobedski, Greenspan and Roth 2001; Tsuda and Nakasuka 2003). An exhaustive list of guidance, navigation & control (GNC) architecture efforts related to rendezvous can be obtained in Nolet (2007). The specific research addressed here currently has direct application to a specific type of problem scenario posed by Nolet (2007) to rendezvous with a target that is “drifting and tumbling” as defined below:

Target drifting or tumbling: it is able to communicate its states, but has no control authority on its displacements and attitude (e.g., docking with a fuel-depleted target with no reaction wheels).

Targets that are fully cooperative or able to communicate and control attitude information would also be candidates as considered by Romano (Romano and Hall 2006; Romano, Friedman and Shay 2007). This research would be coupled with advanced navigation methods to attempt to explore scenarios where there is no information sharing between satellites as listed below (Nolet 2007):

Target not cooperating:

- the target has no control authority, and no communication is occurring (e.g., docking with a dead satellite);

- the target is actively trying to escape the chaser, and no communication is occurring (e.g., military applications).

Regardless of the specific application and overall GNC architecture, the trajectory generation methods will assume to have state information regarding the attitude and position of the tumbling RSO.

This dissertation is organized as follows: Chapter I contains a summary of past and current research on this topic and an outline of several concepts employed in the dissertation. Chapter II focuses on the 6 DoF spacecraft rendezvous problem while formulating the dynamic equations for matching points of interest on two separate spacecraft, one of which is controlled, the other of which is tumbling. The optimal control problem is addressed, resulting in deriving the adjoint equations and transversality conditions needed to verify optimality. Solutions are obtained using a direct method and are verified using indirect shooting methods based on the MP (Pontryagin, Boltyanskii, Gamkrelidze, and Mishchenko 1964). The purpose of this development is to provide a baseline solution to the formulated close approach problem that has not yet been addressed in research. Chapter III expands on the results of Chapter II increasing the complexity of the problem. The same approach is taken, but the translational actuators (thrusters) are body mounted, the inertia matrix of the tumbling object is no longer taken to be identity and the end constraints require not only matching position and velocity of the spacecraft docking points, but also their attitude and angular velocity. Chapter IV introduces the concept of Inverse Dynamics in the Virtual Domain (IDVD) in more detail as it applies to spacecraft reorientation, a key enabler for operations with respect to a tumbling object that has not been addressed so far. Chapter V develops a method for rapid trajectory generation based on IDVD and applies to the spacecraft rendezvous problem. Chapter VI contains analysis and simulations of the trajectories generated using the methods developed. This includes comparing the performance to simplified, 2D cases that exist in literature and evaluating the overall effectiveness when subjected to the tracking challenges of current realistic systems. The chapter concludes with ideas for implementation in closed-loop architecture that would exploit the calculation and robustness advantages of IDVD methods.

A. OPTIMAL CONTROL PROBLEM FORMULATION FOR CLOSE RENDEZVOUS AND DOCKING

Previously, the optimal control for rendezvous with a tumbling object has only been studied in 2D (Ma, Ma and Shashikanth 2007) with very little analysis. Studying the optimal solution of a given problem also entails verifying its optimality. This is done through rigorous analysis of the states, costates and how they relate to the MP. This research formulates the three-dimensional (3D) problem and utilizes a direct collocation method (specifically a pseudospectral method) to obtain a solution, which is then validated using an indirect method based on the MP. Although direct collocation methods enable the generation of an optimal solution (while indirect methods have a very limited radius of convergence), they are extremely computationally intensive and require a significant amount of time to converge (Yakimenko, Xu and Basset 2008, Boyarko, Yakimenko and Romano 2009a and 2009b). Therefore, other rapid-trajectory generation techniques are developed, while using the optimal solution as a baseline for comparison.

B. INVERSE DYNAMICS IN THE VIRTUAL DOMAIN

1. The Concept of Inverse Dynamics in the Virtual Domain

On top of having an extremely large computational time, direct collocation methods may also converge to sub-optimal solution if the number of nodes is not sufficient (Yakimenko et al. 2008; Boyarko et al. 2009a and 2009b). In addition to that, the optimal solution does not have an analytical representation, which may pose problems when trying to implement it using a feed-forward scheme of suggested control commands (Yakimenko et al. 2008; Boyarko et al. 2009a and 2009b). Furthermore, the placement of nodes by a direct collocation method that represent the solution are not flexible can cause problems when trying to interpolate complex control solutions that are not necessarily continuous (Hurni 2009).

This research pursues another approach exploiting the general idea of the direct optimization methods of calculus of variations together with an inverse dynamics approach. In particular, polynomials are used as basis functions to generate trajectories that can be traversed according to a computed analytic speed profile. An abstract

argument is used that allows the trajectory to be decoupled in space and time, while always resulting in a trajectory that will reach the desired endpoint conditions. The resulting quasi-optimal trajectory solution is capable of being generated (and updated) rapidly because of the reduction in the number of varied parameters due to the restriction on the trajectory structure by specifying a polynomial basis. Although this method lacks some flexibility due to the predefined structure, it provides a feasible solution that satisfied the endpoint constraints on the trajectory at every iteration, even when the initial conditions change due to disturbances or delays. Specific applications include scenarios, where derivative conditions on beginning and ending states need to be met, such as tracking missions, docking missions and other missions that are not necessarily rest-to-rest maneuvers. It also has application in areas, where computational time and the ability to immediately employ the best available solution at any given time, is heavily weighted as part of the overall PI of the solution.

2. Attitude Trajectory Generation

The goal of this research is to provide a method to determine a feasible attitude trajectory solution that meets endpoint requirements and dynamic constraints while performing a good overall maneuver relatively to a given cost function. Also, the method should work for any terminal conditions including nonrest to nonrest maneuvers. The major requirement is that the method must focus on providing a rapid, potentially real time solution as opposed to off-line computations even if it requires some sacrifices in optimality. This methods developed here are coupled with methods developed in a later section for the translational trajectory generation, providing the complete solution needed for close approach and docking with a tumbling RSO.

3. Translational Trajectory Generation

Trajectory generation by IDVD has been used in the past for aeronautical applications where the only two frames that needed consideration were the inertial and body frames (Yakimenko et al. 2008; Yakimenko and Slegers 2009). This research utilizes the concept of IDVD in the orbital frame fixed at the RSO (an intermediate frame between the inertial and body frame) and exploiting the Clossey-Wilshire Equations of

relative motion. This allows constructing the problem to analytically set the position and rate values of the states in the specific frame of interest, ensuring a maneuver that will perform the desired close approach maneuver with respect to the RSO, allowing for rapid trajectory reshaping or reoptimizing and reducing overall relative error. The use of a speed factor of a specific form (having an analytical expression for the integral of the inverse function), also allows flexibility in setting the node points at specific time intervals. This new method of rapid-trajectory generation allows the computation of an optimized, feasible, safe trajectory for spacecraft rendezvous requiring significantly less computational time than current methods.

C. SCOPE OF THE DISSERTATION

This dissertation advances the body of knowledge with respect to guidance of a spacecraft to rendezvous and dock with a tumbling object. In particular, the main new contributions of this dissertation are:

1. The analytical formulation of the optimal control problem is presented with respect to a spacecraft performing a controlled approach in 3D to a tumbling object for the purpose of rendezvous and docking. A baseline optimal solution is presented using pseudospectral methods and verified the necessary conditions for optimality based on the Minimum Principle coupled with an indirect shooting method.
2. An analytic quaternion trajectory is formulated using a 5th and 7th order Bezier Polynomial. Derivatives are formulated and applied in the virtual domain providing an attitude trajectory generation technique that can automatically match specified derivative values at the endpoints while varying the speed across the spatial trajectory. This formulation is then applied for rapid trajectory generation, significantly decreasing the computational time.
3. The analytic quaternion trajectory approach based on Bezier polynomials is coupled with a translational polynomial scheme to create a novel trajectory generation scheme for the close approach with a tumbling object. The method is extremely flexible allowing for real time reshaping of the trajectory based

on dynamically changing end constraints. Also, by incorporating a speed factor of a particular form, this method can provide continuous mapping from the virtual and time domains. Analysis shows that a trajectory can be calculated that is within ~10% of the optimal trajectory, but only requiring a fraction (~0.5%) of the computational time.

4. The IDVD method is incorporated in different GNC architecture schemes to demonstrate its flexibility. An IDVD reshaping method is created and implemented in a real time rapid-trajectory generator that incorporates the most current knowledge of the RSO states. A hybrid method, using a combination of recalculating and reoptimizing the solution as well as reshaping based on current RSO state knowledge, is developed and demonstrated.

II. FORMULATION AND ANALYSIS OF MATCHING POINTS OF INTEREST IN TWO-SPACECRAFT FOR OPTIMAL RENDEZVOUS

From the theoretical standpoint, this chapter elaborates on the previous work by Ma (Ma et al. 2007), who has studied the minimum-control effort problem for a planar rendezvous to a tumbling object (with only three states, coordinates x and y , and heading angle θ), neglecting any path constraints and relative motion dynamics pertinent to proximity space operations. The Sakawa-Shindo algorithm was used for calculating the optimal control. This chapter expands the scope by taking into account the proximity motion dynamics, considering the full six DoF model, and determining both the minimum time and the quadratic formulation of minimum-control (energy) effort solution of the rendezvous of two satellites. It also features a comparison of the solutions obtained using one of the prominent direct collocation methods with the truly optimal solutions obtained using the MP. Another section of this dissertation takes an even further step and considers three different PIs, adding additional constraints to match terminal attitude and angular dynamics, along with position and velocity.

A. TWO-SPACECRAFT RENDEZVOUS MODELING AND OPTIMIZATION PROBLEM FORMULATION

This section develops a model of target-chaser rendezvous. Figures 2 and 3 show a graphical representation of the problem. The center of the orbit frame is fixed to the center of mass (CM) of the tumbling RSO. The x axis points toward the Zenith. The y axis lies along the velocity vector of the RSO (assuming circular orbit), and the z axis lies along the orbit normal of the RSO. We start from the arbitrary relative position (Figure 2) and attempt to bring two spacecraft together for docking (Figure 3).

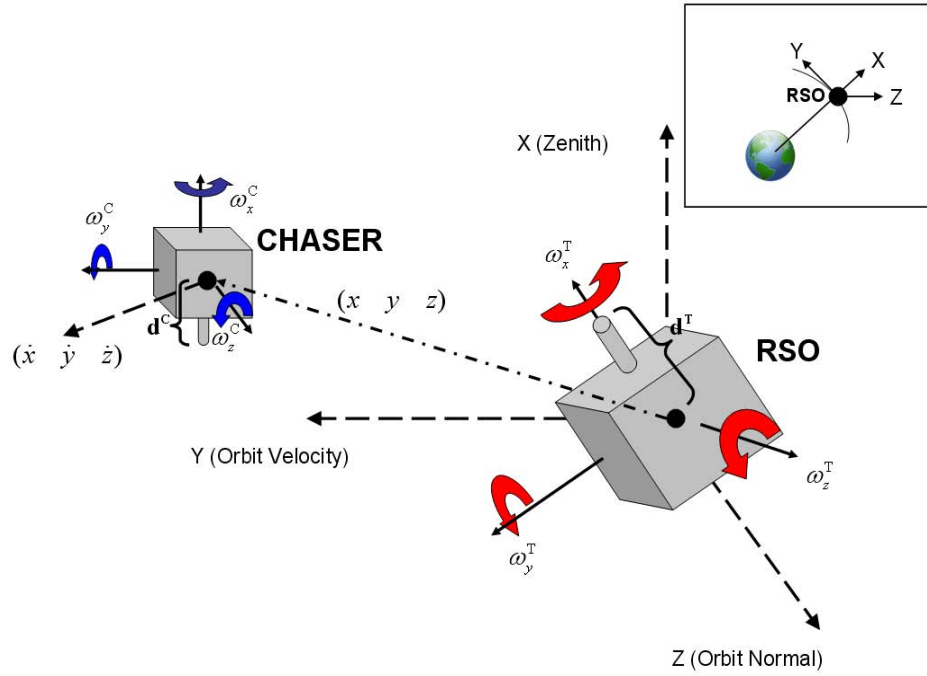


Figure 2. Depiction of the spacecraft rendezvous problem.

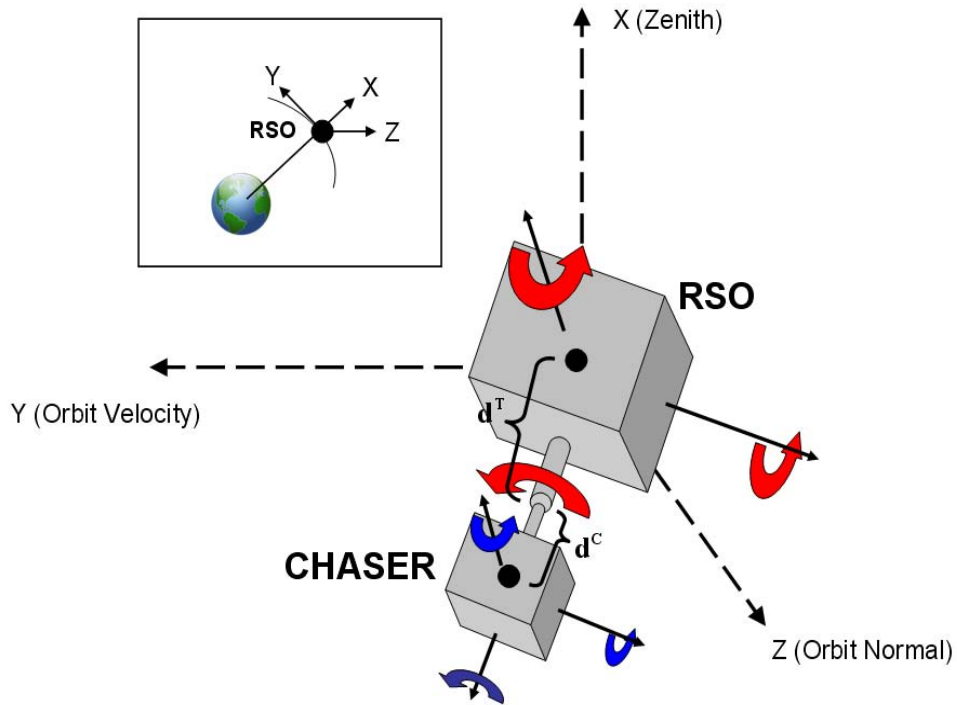


Figure 3. The desired final state of two spacecraft system.

The set of notations to be used in this model (some of which are shown on Figures 2 and 3) include:

m	the mass of the chaser spacecraft;
$d_i^T, i = x, y, z$	the position vector of the target docking point with respect to the target CM expressed in the target body frame;
$d_i^C, i = x, y, z$	the position vector of the chaser docking point with respect to the chaser CM expressed in the chaser body frame;
x, y, z	the x, y , and z position of a chaser spacecraft CM in the Hill's coordinate frame;
$\dot{x}, \dot{y}, \dot{z}$	the x, y , and z component of the velocity of chaser spacecraft center of mass in the Hill's coordinate frame;
$\omega_i^C, i = x, y, z$	the x, y , and z component of the angular velocity of the chaser spacecraft with respect to the inertial frame, expressed in the chaser spacecraft principal body coordinate frame;
$\omega_i^T, i = x, y, z$	the x, y , and z component of the angular velocity of the target spacecraft with respect to the inertial frame, expressed in the target spacecraft principal body coordinate frame;
${}^o\omega_i^C, i = x, y, z$	the x, y , and z component of the angular velocity of the chaser spacecraft with respect to the orbital frame, expressed in the chaser spacecraft principal body coordinate frame;
${}^o\omega_i^T, i = x, y, z$	the x, y , and z component of the angular velocity of the target spacecraft with respect to the orbital frame, expressed in the target spacecraft principal body coordinate frame;
$q_i^C, i = 1, 2, 3, 4$	the components of the quaternion, representing rotation from the orbital to chaser body frame;

$q_i^T, i = 1, 2, 3, 4$	the components of the quaternion, representing rotation from the orbital to target body frame;
Ω	the angular rate of the orbital frame with respect to the inertial frame.

Using these notations, the dynamics of the two systems can now be described as follows (Figure 2). The translational kinematics and dynamics of a chaser spacecraft in the orbit frame centered at the target vehicle are presented by Hill's equations (Vallado and McClain 2007):

$$\begin{aligned}
\ddot{x} &= \frac{1}{m}(2\Omega\dot{y} + 3\Omega^2x + f_x) \\
\ddot{y} &= \frac{1}{m}(-2\Omega\dot{x} + f_y) \\
\ddot{z} &= \frac{1}{m}(-\Omega^2z + f_z)
\end{aligned} \tag{1}$$

where f_x , f_y and f_z are the applied forces (controls) expressed in Hill's frame.

The rotational dynamics of the chaser, described by the vector equation (Greenwood 1987; Wie 1988)

$$\mathbf{I}^C \dot{\boldsymbol{\omega}}^C + \boldsymbol{\omega}^C \times \mathbf{I}^C \boldsymbol{\omega}^C = \mathbf{T} \tag{2}$$

expands into the scalar quantities:

$$\begin{aligned}
\dot{\omega}_x^C &= \frac{(I_{22}^C - I_{33}^C)\omega_y^C \omega_z^C + T_x^C}{I_{11}^C} \\
\dot{\omega}_y^C &= \frac{(I_{33}^C - I_{11}^C)\omega_x^C \omega_z^C + T_y^C}{I_{22}^C} \\
\dot{\omega}_z^C &= \frac{(I_{11}^C - I_{22}^C)\omega_x^C \omega_y^C + T_z^C}{I_{33}^C}
\end{aligned} \tag{3}$$

In Equations (2), (3) $\mathbf{I}^C = \text{diag}([I_{11}^C, I_{22}^C, I_{33}^C])$ is the inertia matrix along the principal axes, $\boldsymbol{\omega}^C = [\omega_x^C, \omega_y^C, \omega_z^C]^T$, and $\mathbf{T} = [T_x, T_y, T_z]^T$ is a vector torques (symbol T denotes transposition).

Similarly, for the target (we have no control of) the rotational dynamics is given by the vector equation

$$\mathbf{I}^T \dot{\boldsymbol{\omega}}^T + \boldsymbol{\omega}^T \times \mathbf{I}^T \boldsymbol{\omega}^T = \mathbf{0} \quad (4)$$

or its scalar form

$$\begin{aligned} \dot{\omega}_x^T &= -\frac{(I_{33}^T - I_{22}^T)\omega_y^T \omega_z^T}{I_{11}^T} \\ \dot{\omega}_y^T &= -\frac{(I_{11}^T - I_{33}^T)\omega_x^T \omega_z^T}{I_{22}^T} \\ \dot{\omega}_z^T &= -\frac{(I_{22}^T - I_{11}^T)\omega_y^T \omega_x^T}{I_{33}^T} \end{aligned} \quad (5)$$

Defining ${}^O_T \mathbf{R}$ as the rotation matrix to convert from the body frame of the target $\{T\}$ to the orbital frame $\{O\}$ and ${}^O_C \mathbf{R}$ to convert from chaser body frame $\{C\}$ to the orbital frame $\{O\}$, we can define the angular velocity of each object ($\alpha = \{T, C\}$) with respect to the orbital frame expressed in the orbital frame:

$$\begin{bmatrix} {}^O \omega_x^\alpha \\ {}^O \omega_y^\alpha \\ {}^O \omega_z^\alpha \end{bmatrix} = {}^O_\alpha \mathbf{R} \begin{bmatrix} \omega_x^\alpha \\ \omega_y^\alpha \\ \omega_z^\alpha \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ \Omega \end{bmatrix} \quad (6)$$

Rotation matrices ${}^O_\alpha \mathbf{R}$ are constructed using components of the corresponding quaternion as follows (Greenwood 1987; Wie 1988):

$${}^O_\alpha \mathbf{R} = \begin{bmatrix} q_4^{\alpha 2} + q_1^{\alpha 2} - q_2^{\alpha 2} - q_3^{\alpha 2} & 2(q_1^\alpha q_2^\alpha - q_3^\alpha q_4^\alpha) & 2(q_1^\alpha q_3^\alpha + q_2^\alpha q_4^\alpha) \\ 2(q_1^\alpha q_2^\alpha + q_3^\alpha q_4^\alpha) & q_4^{\alpha 2} - q_1^{\alpha 2} + q_2^{\alpha 2} - q_3^{\alpha 2} & 2(q_2^\alpha q_3^\alpha - q_1^\alpha q_4^\alpha) \\ 2(q_1^\alpha q_3^\alpha - q_2^\alpha q_4^\alpha) & 2(q_2^\alpha q_3^\alpha + q_1^\alpha q_4^\alpha) & q_4^{\alpha 2} - q_1^{\alpha 2} - q_2^{\alpha 2} + q_3^{\alpha 2} \end{bmatrix} \quad (7)$$

Quaternions are a parameterization for spacecraft attitude transformations based on the fact that the general displacement of a rigid body with one fixed point is a rotation about a fixed axis: the eigenaxis. The definition of the quaternion is:

$$\mathbf{q} = \begin{bmatrix} \rho_1 \sin(\sigma / 2) \\ \rho_2 \sin(\sigma / 2) \\ \rho_3 \sin(\sigma / 2) \\ \cos(\sigma / 2) \end{bmatrix} \quad (8)$$

where σ is the scalar value of the rotational displacement about the eigenaxis and \mathbf{p} is unit vector that describes the direction of the eigenaxis. The parameters of quaternions are propagated according to the known relation (Greenwood 1987; Wie 1988):

$$\begin{bmatrix} \dot{q}_1^\alpha \\ \dot{q}_2^\alpha \\ \dot{q}_3^\alpha \\ \dot{q}_4^\alpha \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & {}^O\omega_z^\alpha & -{}^O\omega_y^\alpha & {}^O\omega_x^\alpha \\ -{}^O\omega_z^\alpha & 0 & {}^O\omega_x^\alpha & {}^O\omega_y^\alpha \\ {}^O\omega_y^\alpha & -{}^O\omega_x^\alpha & 0 & {}^O\omega_z^\alpha \\ -{}^O\omega_x^\alpha & -{}^O\omega_y^\alpha & -{}^O\omega_z^\alpha & 0 \end{bmatrix} \begin{bmatrix} q_1^\alpha \\ q_2^\alpha \\ q_3^\alpha \\ q_4^\alpha \end{bmatrix} \quad (9)$$

Equations (1), (3), (5) and (9) define a 20-state system of differential equations governing rendezvous dynamics. Combined into the state vector \mathbf{x} these states are:

$$\mathbf{x} = [x, y, z, \dot{x}, \dot{y}, \dot{z}, \omega_x^C, \omega_y^C, \omega_z^C, \omega_x^T, \omega_y^T, \omega_z^T, q_1^C, q_2^C, q_3^C, q_4^C, q_1^T, q_2^T, q_3^T, q_4^T]^T \quad (10)$$

The governing dynamics assume six normalized controls that can be used by the chaser to achieve the rendezvous conditions:

$$\mathbf{u} = \left[\frac{f_x}{f_{x\max}}, \frac{f_y}{f_{y\max}}, \frac{f_z}{f_{z\max}}, \frac{T_x}{T_{x\max}}, \frac{T_y}{T_{y\max}}, \frac{T_z}{T_{z\max}} \right]^T \quad (11)$$

For simplicity, we will further assume $f_{i\max} = 1m/s^2$, $T_{i\max} = 1Nm$, $i = x, y, z$. These controls are three normalized components of a translational force acting on the chaser f_i , $i = x, y, z$, expressed in the Hill's coordinate frame, and three normalized

components of a torque allowing to change chaser's attitude, T_i , $i = x, y, z$, expressed in the chaser's body frame.) All six controls are bounded: $-\mathbf{1} \leq \mathbf{u} \leq \mathbf{1}$.

Using these controls, we would like to bring the two spacecraft from some initial conditions, given by 20 initial values of states $x_i(t_0)$, $i = 1, \dots, 20$, to docking-enabling conditions described by matching chaser's and target's docking station final positions and velocity vectors as described by the following equations. These conditions for positions and velocities, expressed in the matrix form are as follows:

$${}^o_T\mathbf{R} \begin{bmatrix} d_x^T \\ d_y^T \\ d_z^T \end{bmatrix} - \left({}^o_C\mathbf{R} \begin{bmatrix} d_x^C \\ d_y^C \\ d_z^C \end{bmatrix} + \begin{bmatrix} x \\ y \\ z \end{bmatrix} \right) = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \mathbf{0} \quad \text{and}$$

$${}^o\boldsymbol{\omega}^T \times {}^o_T\mathbf{R} \begin{bmatrix} d_x^T \\ d_y^T \\ d_z^T \end{bmatrix} - \left({}^o\boldsymbol{\omega}^C \times {}^o_C\mathbf{R} \begin{bmatrix} d_x^C \\ d_y^C \\ d_z^C \end{bmatrix} + \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \right) = \begin{bmatrix} e_4 \\ e_5 \\ e_6 \end{bmatrix} = \mathbf{0} \quad (12)$$

While transitioning to docking-enabling conditions, we would like to minimize and compare two different performance indices:

$$J = \int_{t_0}^{t_f} f_0 dt \quad (13)$$

with

$$f_0 = 1 \quad (14)$$

for minimum time and

$$f_0 = \frac{1}{2} (u_1^2 + u_2^2 + u_3^2 + u_4^2 + u_5^2 + u_6^2) \quad (15)$$

for minimum quadratic-control (or more commonly referred to as minimum energy) expenditure (Henshaw 2003). An equivalent formulation to the minimum time cost can also be expressed as $J = t_f$, which employs an endpoint cost (Mayer) as opposed to a

running cost (Lagrange). A brief statement about the cost function (or PI) is in order. The specific cost function for minimum energy was chosen to include all actuators. The fact is, it can contain any subset of the controls or states. The choice for this research was to keep the cost function as generic as possible (including all actuator in the cost), but one can easily think of situations, where the cost function might be truncated to only include translational actuators (the case of using Control moment Gyroscopes (CMG) or Reaction Wheels (RW) instead of thrusters for attitude control). The specific cost function implemented would depend on the details of the mission, system and the discretion of the user, regardless of the method applied to obtain a solution.

B. SYNTHESIS OF THE OPTIMAL CONTROL USING MINIMUM PRINCIPLE

1. Formulation of the Optimal Control Problem

We start from the general formulation for the Hamiltonian of the system with the state vector \mathbf{x} , Equation (10), controls vector \mathbf{u} , Equation (11), and running cost of Equations (13)-(15):

$$H(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{u}) := f_0 + (\boldsymbol{\lambda}, \dot{\mathbf{x}}) \quad (16)$$

where operator (\dots) on the right-hand side denotes a scalar product of two vectors, and $\boldsymbol{\lambda} \in \mathbb{R}^{N_x}$ is a costate vector which differential equations are to be defined later in this section.

For the specific system of Equations (1), (3), (5) and (9), with the running cost from Equation (14) (minimum time problem) the Hamiltonian can be written with respect to the state vector \mathbf{x} defined in Equation (10) as:

$$\begin{aligned}
H(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{u}) := & 1 + \lambda_1 x_4 + \lambda_2 x_5 + \lambda_3 x_6 + \frac{1}{m} \left(\lambda_4 (2\Omega x_5 + 3\Omega^2 x_1 + u_1) + \lambda_5 (-2\Omega x_4 + u_2) + \lambda_6 (-\Omega^2 x_3 + u_3) \right) \\
& + \frac{(I_{22}^C - I_{33}^C) x_8 x_9 + u_4}{I_{11}^C} \lambda_7 + \frac{(I_{33}^C - I_{11}^C) x_7 x_9 + u_5}{I_{22}^C} \lambda_8 + \frac{(I_{11}^C - I_{22}^C) x_8 x_7 + u_6}{I_{33}^C} \lambda_9 \\
& + \frac{(I_{22}^T - I_{33}^T) x_{11} x_{12}}{I_{11}^T} \lambda_{10} + \frac{(I_{33}^T - I_{11}^T) x_{10} x_{12}}{I_{22}^T} \lambda_{11} + \frac{(I_{11}^T - I_{22}^T) x_{11} x_{10}}{I_{33}^T} \lambda_{12} \\
& + \lambda_{13} \frac{1}{2} \left(\left(x_9 - (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2) \Omega \right) x_{14} - \left(x_8 - 2(x_{14} x_{15} - x_{13} x_{16}) \Omega \right) x_{15} + \left(x_7 - 2(x_{13} x_{15} + x_{14} x_{16}) \Omega \right) x_{16} \right) \\
& + \lambda_{14} \frac{1}{2} \left(- \left(x_9 - (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2) \Omega \right) x_{13} + \left(x_8 - 2(x_{14} x_{15} - x_{13} x_{16}) \Omega \right) x_{16} + \left(x_7 - 2(x_{13} x_{15} + x_{14} x_{16}) \Omega \right) x_{15} \right) \\
& + \lambda_{15} \frac{1}{2} \left(- \left(x_7 - 2(x_{13} x_{15} + x_{14} x_{16}) \Omega \right) x_{14} + \left(x_8 - 2(x_{14} x_{15} - x_{13} x_{16}) \Omega \right) x_{13} + \left(x_9 - (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2) \Omega \right) x_{16} \right) \\
& + \lambda_{16} \frac{1}{2} \left(- \left(x_7 - 2(x_{13} x_{15} + x_{14} x_{16}) \Omega \right) x_{13} - \left(x_8 - 2(x_{14} x_{15} - x_{13} x_{16}) \Omega \right) x_{14} - \left(x_9 - (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2) \Omega \right) x_{15} \right) \\
& + \lambda_{17} \frac{1}{2} \left(\left(x_{12} - (x_{20}^2 - x_{17}^2 - x_{18}^2 + x_{19}^2) \Omega \right) x_{18} - \left(x_{11} - 2(x_{18} x_{19} - x_{17} x_{20}) \Omega \right) x_{19} + \left(x_{10} - 2(x_{17} x_{19} + x_{18} x_{20}) \Omega \right) x_{20} \right) \\
& + \lambda_{18} \frac{1}{2} \left(- \left(x_{12} - (x_{20}^2 - x_{17}^2 - x_{18}^2 + x_{19}^2) \Omega \right) x_{17} + \left(x_{11} - 2(x_{18} x_{19} - x_{17} x_{20}) \Omega \right) x_{20} + \left(x_{10} - 2(x_{17} x_{19} + x_{18} x_{20}) \Omega \right) x_{19} \right) \\
& + \lambda_{19} \frac{1}{2} \left(- \left(x_{10} - 2(x_{17} x_{19} + x_{18} x_{20}) \Omega \right) x_{18} + \left(x_{11} - 2(x_{18} x_{19} - x_{17} x_{20}) \Omega \right) x_{17} + \left(x_{12} - (x_{20}^2 - x_{17}^2 - x_{18}^2 + x_{19}^2) \Omega \right) x_{20} \right) \\
& + \lambda_{20} \frac{1}{2} \left(- \left(x_{10} - 2(x_{17} x_{19} + x_{18} x_{20}) \Omega \right) x_{17} - \left(x_{11} - 2(x_{18} x_{19} - x_{17} x_{20}) \Omega \right) x_{18} - \left(x_{12} - (x_{20}^2 - x_{17}^2 - x_{18}^2 + x_{19}^2) \Omega \right) x_{19} \right)
\end{aligned} \tag{17}$$

The part of the Hamiltonian that depends on the controls, the switching function, for the time-minimum problem is:

$$H^*(\lambda, \mathbf{x}, \mathbf{u}) = \frac{1}{m}(\lambda_4 u_1 + \lambda_5 u_2 + \lambda_6 u_3) + \frac{1}{I_{11}^C} \lambda_7 u_4 + \frac{1}{I_{22}^C} \lambda_8 u_5 + \frac{1}{I_{33}^C} \lambda_9 u_6 \quad (18)$$

As all six controls enter the switching function (Hamiltonian) linearly, the optimal control for all of them is the bang-bang control defined by:

$$u_i = \begin{cases} 1, & \text{when } \lambda_{i+3} < 0 \\ -1, & \text{when } \lambda_{i+3} \geq 0 \end{cases} \quad \text{for } i = 1, \dots, 6 \quad (19)$$

(the possibility of a singular control, when $\lambda_{i+3}(t) \equiv 0$, is considered in Chapter II.B.3).

Likewise, developing the Hamiltonian for the minimum quadratic-control cost function, based on Equation (15) the part of the Hamiltonian that depends on the controls is:

$$\begin{aligned} H^*(\lambda, \mathbf{x}, \mathbf{u}) := & \frac{1}{2}(u_1^2 + u_2^2 + u_3^2 + u_4^2 + u_5^2 + u_6^2) \\ & + \frac{1}{m}(\lambda_4 u_1 + \lambda_5 u_2 + \lambda_6 u_3) + \frac{1}{I_{11}^C} \lambda_7 u_4 + \frac{1}{I_{22}^C} \lambda_8 u_5 + \frac{1}{I_{33}^C} \lambda_9 u_6 \end{aligned} \quad (20)$$

Since the controls enter Hamiltonian nonlinearly, the optimal control is not the bang-bang control anymore. To be more specific, the resulting optimal control that minimizes the Hamiltonian for the minimum quadratic-control cost function Equation (15) (minimum energy) is as follows:

$$\begin{aligned}
u_i &= \begin{cases} 1, & \frac{\lambda_{i+3}}{m} < -1 \\ -\frac{\lambda_{i+3}}{m}, & -1 < \frac{\lambda_{i+3}}{m} < 1 \\ -1, & \frac{\lambda_{i+3}}{m} \geq 1 \end{cases} \quad \text{for } i = 1, 2, 3, \\
u_i &= \begin{cases} 1, & \frac{\lambda_{i+3}}{I_{kk}^C} < -1 \\ -\frac{\lambda_{i+3}}{I_{kk}^C}, & -1 < \frac{\lambda_{i+3}}{I_{kk}^C} < 1 \\ -1, & \frac{\lambda_{i+3}}{I_{kk}^C} \geq 1 \end{cases} \quad \text{for } i = 4, 5, 6, \text{ where } k = i - 3
\end{aligned} \tag{21}$$

Now, the differential equations for costates will be the same for both optimization problems and are obtained via:

$$\dot{\lambda} = - \left(\frac{\partial H}{\partial \mathbf{x}} \right)^T \tag{22}$$

The first six adjoint equations are given by:

$$\begin{aligned}
\dot{\lambda}_1 &= -3\lambda_4\Omega^2 m^{-1}, \quad \dot{\lambda}_2 = 0, \quad \dot{\lambda}_3 = \lambda_6\Omega^2 m^{-1}, \\
\dot{\lambda}_4 &= -\lambda_1 + 2\lambda_5\Omega m^{-1}, \quad \dot{\lambda}_5 = -\lambda_2 - 2\lambda_4\Omega m^{-1}, \quad \dot{\lambda}_6 = -\lambda_3
\end{aligned} \tag{23}$$

The next six adjoint equations, corresponding to the states 7 through 12, take the form of:

$$\begin{aligned}
\dot{\lambda}_7 &= \frac{I_{11}^C - I_{33}^C}{I_{22}^C} \lambda_8 x_9 + \frac{I_{22}^C - I_{11}^C}{I_{33}^C} \lambda_9 x_8 - \lambda_{13} \frac{1}{2} \Omega x_{16} \\
&\quad - \lambda_{14} \frac{1}{2} \Omega x_{15} + \lambda_{15} \frac{1}{2} \Omega x_{14} + \lambda_{16} \frac{1}{2} \Omega x_{13}
\end{aligned} \tag{24}$$

The remaining eight adjoint equations, corresponding to the states 13 through 20, are of the form:

$$\begin{aligned}
\dot{\lambda}_{13} = & -\lambda_{13} \frac{1}{2} (2x_{13}\Omega x_{14} - 2x_{16}\Omega x_{15} - 2x_{15}\Omega x_{16}) \\
& -\lambda_{14} \frac{1}{2} \left(-\left(x_9 - (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2) \right) \Omega \right) - 2x_{13}^2 \Omega - 2x_{16}^2 \Omega - 2x_{15}^2 \Omega \\
& -\lambda_{15} \frac{1}{2} \left(2x_{15}\Omega x_{14} + (x_8 - 2(x_{14}x_{15} - x_{13}x_{16})) \Omega \right) + 2x_{16}\Omega x_{13} + 2x_{13}\Omega x_{16} \\
& -\lambda_{16} \frac{1}{2} \left(2x_{15}\Omega x_{13} - (x_7 - 2(x_{13}x_{15} + x_{14}x_{16})) \Omega \right) - 2x_{16}\Omega x_{14} - 2x_{13}\Omega x_{15} \\
\\
\dot{\lambda}_{14} = & -\lambda_{13} \frac{1}{2} \left(2x_{14}^2 \Omega + \left(x_9 - (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2) \right) \Omega \right) + 2x_{16}^2 \Omega + 2x_{15}^2 \Omega \\
& -\lambda_{14} \frac{1}{2} (2x_{13}\Omega x_{14} - 2x_{16}\Omega x_{15} - 2x_{15}\Omega x_{16}) \\
& -\lambda_{15} \frac{1}{2} \left(-\left(x_7 - 2(x_{13}x_{15} + x_{14}x_{16}) \right) \Omega \right) + 2x_{16}\Omega x_{14} - 2x_{15}\Omega x_{13} + 2x_{14}\Omega x_{16} \\
& -\lambda_{16} \frac{1}{2} \left(2x_{16}\Omega x_{13} - (x_8 - 2(x_{14}x_{15} - x_{13}x_{16})) \Omega \right) + 2x_{15}\Omega x_{14} - 2x_{14}\Omega x_{15} \\
\\
\dot{\lambda}_{15} = & -\lambda_{13} \frac{1}{2} \left(-2x_{15}\Omega x_{14} - (x_8 - 2(x_{14}x_{15} - x_{13}x_{16})) \Omega \right) + 2x_{15}\Omega x_{14} - 2x_{13}\Omega x_{16} \\
& -\lambda_{14} \frac{1}{2} \left(2x_{15}\Omega x_{13} - 2x_{16}\Omega x_{14} + (x_7 - 2(x_{13}x_{15} + x_{14}x_{16})) \Omega \right) - 2x_{13}\Omega x_{15} \\
& -\lambda_{15} \frac{1}{2} (2x_{13}\Omega x_{14} - 2x_{14}\Omega x_{13} + 2x_{15}\Omega x_{16}) \\
& -\lambda_{16} \frac{1}{2} \left(+2x_{13}^2 \Omega + 2x_{14}^2 \Omega - \left(x_9 - (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2) \right) \Omega \right) + 2x_{15}^2 \Omega \\
\\
\dot{\lambda}_{16} = & -\lambda_{13} \frac{1}{2} \left(-2x_{16}\Omega x_{14} - 2x_{13}\Omega x_{15} + (x_7 - 2(x_{13}x_{15} + x_{14}x_{16})) \Omega \right) - 2x_{16}\Omega x_{14} \\
& -\lambda_{14} \frac{1}{2} \left(2x_{16}\Omega x_{13} + 2x_{16}\Omega x_{13} + (x_8 - 2(x_{14}x_{15} - x_{13}x_{16})) \Omega \right) - 2x_{14}\Omega x_{15} \\
& -\lambda_{15} \frac{1}{2} \left(2x_{13}^2 \Omega + 2x_{14}^2 \Omega - \left(x_9 - (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2) \right) \Omega \right) + 2x_{16}^2 \Omega \\
& -\lambda_{16} \frac{1}{2} (2x_{13}\Omega x_{14} - 2x_{13}\Omega x_{14} + 2x_{15}\Omega x_{16})
\end{aligned} \tag{25}$$

(The adjoint equations for $\dot{\lambda}_{17}, \dots, \dot{\lambda}_{20}$ can be derived by swapping the associated chaser states and costates with the RSO variables (ex. $\dot{\lambda}_{13} \rightarrow \dot{\lambda}_{17}, x_{13} \rightarrow x_{17}, \lambda_{13} \rightarrow \lambda_{17}$, etc.)).

For this problem formulation, it should also be noted that the third and the sixth costate equations are decoupled. This feature will be addressed further. The next step is to examine the transversality conditions, which with account to the following coupled terminal variations, followed directly from Equation (12). The six scalar equations for the components of the vector $\mathbf{e}=[e_1,...,e_6]'$ represent the matching of position and velocity of the docking points in the three components of the orbital coordinate frame at the time t_f . If positions of the docking points are matched in the orbital frame, then matching velocity in the orbital frame will lead to matching velocity in the inertial frame as well, since the transport theorem will have the same effect for coincident points.

Let variable E represent an endpoint cost that is set to zero for both the minimum time and minimum-control scenarios. Introducing an auxiliary function, \bar{E} , several more necessary conditions for optimal control (i.e., transversality conditions) can be taken into account as (Yan, Fahroo and Ross 2002):

$$\bar{E} = E + \sum_{i=1}^6 \nu_i e_i \quad (26)$$

$$\left(\frac{\partial \bar{E}}{\partial \mathbf{x}} \right)^T \bigg|_{t=t_f} = \boldsymbol{\lambda}_{t_f} \quad (27)$$

The first six equations result in expressions that only contain the parameters ν_i , $i=1,...,6$:

$$\begin{aligned} \lambda_1(t_f) &= -\nu_1, \lambda_2(t_f) = -\nu_2, \lambda_3(t_f) = -\nu_3, \\ \lambda_4(t_f) &= -\nu_4, \lambda_5(t_f) = -\nu_5, \lambda_6(t_f) = -\nu_6 \end{aligned} \quad (28)$$

These six equations are then properly substituted into the remaining equations to give the 14 equations for the end conditions, described as $\boldsymbol{\zeta}=[\zeta_1,...,\zeta_{14}]'$, that must be satisfied, along with the six docking states for position and velocity given by Equation (12) at t_f ,

and the twenty known initial conditions. For example, for the value at t_f of the 7th costate in relation to the first transversality condition is:

$$\begin{aligned}\lambda_7(t_f) = & \left(2(x_{13}x_{14} + x_{15}x_{16})x_{23} - 2(x_{13}x_{15} - x_{14}x_{16})x_{22}\right)v_4 \\ & + \left(2(x_{13}x_{15} - x_{14}x_{16})x_{21} - (x_{16}^2 + x_{13}^2 - x_{14}^2 - x_{15}^2)x_{23}\right)v_5 \\ & + \left((x_{16}^2 + x_{13}^2 - x_{14}^2 - x_{15}^2)x_{22} - 2(x_{13}x_{14} + x_{15}x_{16})x_{21}\right)v_6\end{aligned}\quad (29)$$

Then substituting from Equation (28) leads to:

$$\begin{aligned}\zeta_1(t_f) = & \lambda_7(t_f) - \left(2(x_{13}x_{14} + x_{15}x_{16})x_{23} - 2(x_{13}x_{15} - x_{14}x_{16})x_{22}\right)(-\lambda_4(t_f)) \\ & + \left(2(x_{13}x_{15} - x_{14}x_{16})x_{21} - (x_{16}^2 + x_{13}^2 - x_{14}^2 - x_{15}^2)x_{23}\right)(-\lambda_5(t_f)) \\ & + \left((x_{16}^2 + x_{13}^2 - x_{14}^2 - x_{15}^2)x_{22} - 2(x_{13}x_{14} + x_{15}x_{16})x_{21}\right)(-\lambda_6(t_f)) = 0\end{aligned}\quad (30)$$

The same is done for the remaining 13 equations.

Therefore, everything is in place to solve the minimum time Minimum Principle problem converted to the nonlinear programming, two point boundary value problem (TPBVP) numerically. Specifically, given the dynamics of the system described by Equations (1), (3), (5), (9), and adjoint system Equations (23)-(25), with the optimal controls synthesized as Equation (19) or Equation (21), with the initial states $x_i(t_0)$, $i=1,\dots,20$, we guess on the values of the initial costates $\lambda_i(t_0)$, $i=1,\dots,20$ and the maneuver time t_f (21 variable parameters), in attempt to zero six final discrepancies in matching final position and inertial velocity of chaser's and target's docking stations (Equation (12)), together with fourteen relations resulting from the transversality conditions (Equation(26)) and assure that (Yan et al. 2002)

$$H(t_f) = 0. \quad (31)$$

Therefore, we have 21 varied parameters and 21 conditions to satisfy.

2. Accounting for a Possible Path Constraint

For the close rendezvous problem, we should obviously take into account one more constraint. This constraint is that the CM of the chaser spacecraft must remain at a

distance larger than some minimum distance (a “keep out” sphere with a radius r) from the CM of the RSO, which is coincident with the origin of the orbit frame. This assures that the chaser vehicle will not pass through the target vehicle in order to reach the docking position. Mathematically, this can be defined as:

$$h = (x_1^2 + x_2^2 + x_3^2) - r^2 \geq 0 \quad (32)$$

where x_i , $i = 1, 2, 3$ are the first three elements of the state vector (9). Furthermore, while a trajectory is on a path constraint $h = 0$, the tangential condition must also be satisfied (Pontryagin et al. 1964):

$$\dot{h} = \frac{dh}{dt} = x_1 x_4 + x_2 x_5 + x_3 x_6 = 0 \quad (33)$$

Consequently, the Hamiltonian should be augmented by another term:

$$H(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{u}) := f_0 + (\boldsymbol{\lambda}, \dot{\mathbf{x}}) + \mu \ddot{h} \quad (34)$$

where μ is a constant and

$$\ddot{h} = \frac{d^2 h}{dt^2} \quad (35)$$

(since the path constraint has to be differentiated with respect to time twice before the control variables appear in the expression). The value of μ is dictated as follows:

$$\begin{aligned} \mu = 0 & \quad \text{if } h > 0 & \Rightarrow & \text{off the constraint boundary} \\ \mu \geq 0 & \quad \text{if } h = \dot{h} = \ddot{h} = 0 & \Rightarrow & \text{on the constraint boundary} \end{aligned} \quad (36)$$

Therefore, upon first contact with the path constraint, the costate values and Hamiltonian will be discontinuous (Bryson and Ho 1975).

3. Possibility of a Singular Control for a Minimum Time Problem

Upon closer inspection, we find that in the z -direction translational control u_3 is decoupled from all other controls. From Equation (1) we have

$$\dot{x}_3 = x_6, \quad \dot{x}_6 = m^{-1}(-\Omega^2 x_3 + u_3) \quad (37)$$

or

$$\ddot{x}_3 + m^{-1}\Omega^2 x_3 = m^{-1}u_3 \quad (38)$$

and Equation (23) yield

$$\dot{\lambda}_3 = m^{-1}\Omega^2 \lambda_6, \quad \dot{\lambda}_6 = -\lambda_3 \quad (39)$$

or

$$\ddot{\lambda}_6 + m^{-1}\Omega^2 \lambda_6 = 0. \quad (40)$$

Now, for the minimum time problem, the optimal control is defined as in Equation (19), hence for u_3 we will have

$$u_3 = \begin{cases} 1, & \lambda_6 < 0 \\ -1, & \lambda_6 > 0 \end{cases} \quad (41)$$

With account of Equation (40), we can state that

$$u_3(t) = f(\lambda_3(t_0), \lambda_6(t_0)). \quad (42)$$

Defined by the natural frequency of Equation (40) the control u_3 can switch from $u_{3\max} = 1$ to $u_{3\min} = -1$ or vice versa every $\pi\sqrt{m}\Omega^{-1}$ seconds. Moreover, the solution to Equation (40) with the initial conditions defined by $\lambda_6(t_0)$ and $\dot{\lambda}_6(t_0) = -\lambda_3(t_0)$ can be found analytically:

$$\lambda_6(t) = -\lambda_3(t_0)m^{0.5}\Omega^{-1}\sin(m^{-0.5}\Omega t) + \lambda_6(t_0)\cos(m^{-0.5}\Omega t). \quad (43)$$

The roots of this equation are defined as

$$t = \tan^{-1} \left(\frac{\lambda_6(t_0)}{\lambda_3(t_0)} m^{-0.5} \Omega \right) m^{0.5} \Omega^{-1}. \quad (44)$$

From the other hand, the extremals of Equation (43) are achieved at

$$t^* = -\tan^{-1} \left(\frac{\lambda_3(t_0)}{\lambda_6(t_0)} m^{0.5} \Omega^{-1} \right) m^{0.5} \Omega^{-1}. \quad (45)$$

As seen from Equations (43)–(45), the only possibility for the singular control would be when

$$\lambda_3(t_0) = 0 \text{ and } \lambda_6(t_0) = 0. \quad (46)$$

In this case, $\lambda_6(t) = 0$ and the optimal control cannot be defined from Equation (41), but requires more rigorous analysis (Bryson and Ho 1975).

C. METHODOLOGY FOR OBTAINING A SOLUTION AND CHECKING THE OPTIMALITY

This section presents the methodology for obtaining and verifying optimal solutions for two problems posted in Chapter II.A. Despite the fact that the structure of an optimal control was defined analytically (in the previous section), it is clear that there is no way one could solve this problem using direct shooting approach having some arbitrary values of varied parameters. No matter what optimization algorithm to solve the TPBVP would be used, the numerical solution will diverge. That is where direct methods of calculus of variations become useful. In what follows, this section introduces a specific rendezvous scenario with the concrete specific numerical values to play with. Next, it describes a procedure of obtaining quasi-optimal numerical solutions for each of two optimization problems using direct collocation methods, also known as pseudospectral methods. Finally, a methodology of using this solution, which is very close to the true optimal one, to address the problem using direct shooting method for the TPBVP formulated in Chapter II.B.1 is introduced.

1. Defining a Rendezvous Scenario

This specific scenario to play with initializes the chaser CM starting at a distance of 5 meters from the target CM with the target having an initial angular velocity of 0.25 rad/s in both y and z body axes without losing generality. The body coordinate frames of each spacecraft and the orbit frame are assumed to be coincident with the inertial frame at the beginning of the simulation. The chaser docking point is located at $[-0.25, 0, 0]^T$ in the body frame while the target docking point is located $[1, 0, 0]^T$ in the target body frame. The initial values of mentioned and the remaining states for computer simulations discussed in the following sections are presented in Table 1. The scenario also assumes $m = 1\text{kg}$, $\Omega = 0.005\text{ rad/s}$, $\mathbf{I}^C = \mathbf{I}^T = \mathbf{I}_{3 \times 3}$ (where $\mathbf{I}_{3 \times 3}$ is the identity matrix), $r^2 = 1.5\text{m}$ and the maximum bound on the final time of 10 seconds. The choice of an Ω value was made arbitrarily.

<i>State</i>	<i>Initial condition (m and m/s)</i>	<i>State</i>	<i>Initial condition (rad/s)</i>	<i>State</i>	<i>Initial condition (quaternion)</i>	<i>State</i>	<i>Initial condition (quaternion)</i>
x_1	0	x_7	0	x_{13}	0	x_{17}	0
x_2	5	x_8	0	x_{14}	0	x_{18}	0
x_3	0	x_9	0	x_{15}	0	x_{19}	0
x_4	0	x_{10}	0	x_{16}	1	x_{20}	1
x_5	0	x_{11}	0.25				
x_6	0	x_{12}	0.25				

Table 1. The initial values of the states

2. Solving the Problem with the Gauss Pseudospectral Optimization Solver (GPOPS)

The optimal control problems posted in Section A were first solved using the Gauss Pseudospectral Optimization Solver (GPOPS) (Rao et al. 2009). The GPOPS is an open source code that implements the Gauss Pseudospectral Method for solving optimal control problems. As many as 150 internal nodes were chosen for the solution. Usually, to speed up the numerical procedure, not more than about 60 nodes are being used (Yakimenko, Xu and Basset 2008). The initial conditions were chosen based on Table 1 and the final conditions were based on matching position and velocity of the terminal point.

3. Verification of the GPOPS Solution with Minimum Principle

Differential equations for the states (Equations (1), (3), (5), (9)) and costates (Equations (22)–(25)), developed in the previous sections, were programmed into the Mathworks' Simulink blocks as shown in Figure 4.

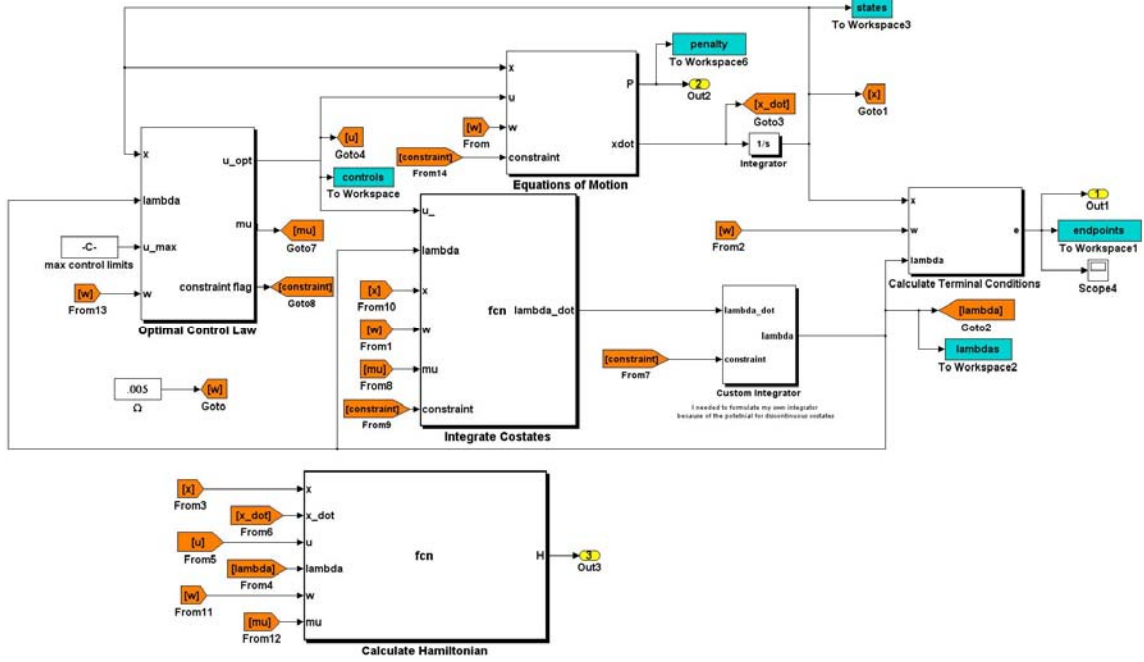


Figure 4. Simulink model for integrating the states and costates using the optimal switching function for controls.

The Optimal Control Law block (on the left in Figure 4), implements the control based on the switching conditions Equation (19) or Equation (21) developed in Chapter II.B.1. The upper and middle blocks integrate the states and costates, respectively, while block on the bottom calculates the Hamiltonian. The initial conditions for the 20 states are taken from Table 1, while the 20 initial conditions for costates along with the final time t_f are guessed on.

The far right block calculates the terminal conditions, \mathbf{e} and $\boldsymbol{\zeta}$. The outputs are sent to a cost function β constructed as follows:

$$\beta = \|\mathbf{e}(t_f)\| + \|\boldsymbol{\zeta}(t_f)\| + H(t_f)^2 \quad (47)$$

The MATLAB unconstrained optimization function *fminunc* (exploiting Quasi-Newton method) was used to solve for the TPBVP in the following form:

$$\mathbf{X} = \text{fminunc}(\text{@cost_function}, \mathbf{x}_0, \text{options}, \text{lambda}, \text{time}).$$

This function attempts to find a minimum of a scalar function of several variables, *cost_function*, starting at an initial estimate \mathbf{x}_0 . Here, *cost_function* calls the Simulink model (Figure 4) passes the vector of initial guesses

$$\mathbf{X} = [\lambda_1(t_0), \dots, \lambda_{20}(t_0), t_f]^T \quad (48)$$

and computes the cost function β . The *options* parameter of *fminunc* is varied based on the fidelity of solution required. By default, the termination tolerance the function value and on vector of varied parameters is set to 10^{-6} .

For the case of path constraints placed on the state variables as in Equation (31), 20 additional parameters are needed to define the costates values at the time, t_d , where they become discontinuous (Bryson and Ho 1975). For this case, a reset is included along with the integrator that will reset the costates to these new parameters if the path constraint is contacted. In this case, the \mathbf{X} value is augmented to include initial guesses of the costate values for the time t_d (Bryson and Ho 1975) and is

$$\mathbf{X} = [\lambda_1(t_0), \dots, \lambda_{20}(t_0), \lambda_1(t_d), \dots, \lambda_{20}(t_d), t_f]^T \quad (49)$$

The vector \mathbf{X} is only augmented with enough costate reset values as suspected path constraint contact points deduced from the GPOPS solution. For example, the GPOPS solution suggests only one contact with the path constraint, therefore, we only augment the \mathbf{X} vector with one set of $\lambda(t_d)$ values.

To implement the state path constraint of the form $h(\mathbf{x}) \geq 0$, where h does not depend on \mathbf{u} , a penalty function P , was associated with the violation of the constraint that took the form of:

$$\rho = \begin{cases} 2x_1x_4 + 2x_2x_5 + 2x_3x_6, & h \leq 0 \\ 0, & h > 0 \end{cases} \quad (50)$$

with

$$P = \int_{t_0}^{t_f} \rho dt \quad (51)$$

and

$$\beta = \|e(t_f)\| + \|\zeta(t_f)\| + H(t_f)^2 + P \quad . \quad (52)$$

instead of Equation (47). This increases the cost function if the vehicle is on the constraint boundary and not meeting the tangency conditions stated in Chapter II.B.2. Otherwise, if the vehicle is not on the constraint or meets the tangency conditions while on the constraint, there is no penalty associated with the cost. A penalty function of this type is appealing because it has a smooth transition from solutions where constraints are not violated to solutions where they are violated. Note, that if the vehicle is on the boundary and meets the tangency conditions, it will not cross the boundary.

D. OBTAINING AND ANALYZING MINIMUM QUADRATIC-CONTROL (MINIMUM ENERGY) SOLUTION

The results of the GPOPS solution and then their verification with the Minimum Principle solution are obtained as discussed in the previous section.

1. Minimum Energy Solution with GPOPS

For the minimum energy rendezvous scenario set in Chapter II.A, the GPOPS optimization package yielded the solution shown in Figure 5. This solution, minimizing the quadratic-control cost returned a value of $J = 0.1133$. Figure 6 shows the planar views of the solution. The overlaid sphere is centered at the target RSO and has a radius equal to that of the distance the docking point of the RSO is offset from its center of gravity. The final maneuver time is calculated to be 10 seconds, the upper bound on the final allowable time for this scenario (without this limit the optimal solution would yield

an infinite final time). It should be noted that although position and velocities of the docking points coincide, the orientation of the spacecraft do not match since this condition was not set as a constraint. In essence, a spacecraft can be translating at a different rate than the docking point because the docking point is offset from the spacecraft CM and rotating with respect to the orbital frame.

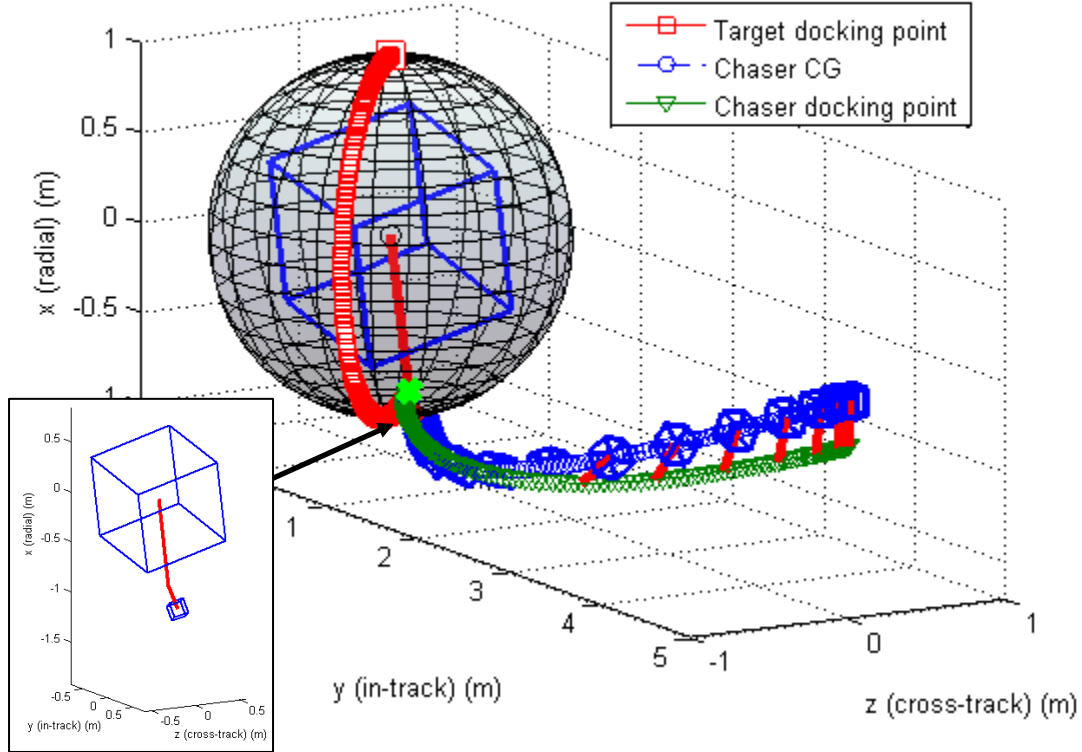


Figure 5. Minimum energy solution (GPOPS): The 3D view of the optimal trajectory. A close-up of the final position is shown in the exploded view.

The progression of the RSO docking point position curves out a circle in the plane perpendicular to the angular velocity vector of the RSO because the RSO is assumed to have an identity inertia matrix, an assumption that is relaxed in the next chapter.

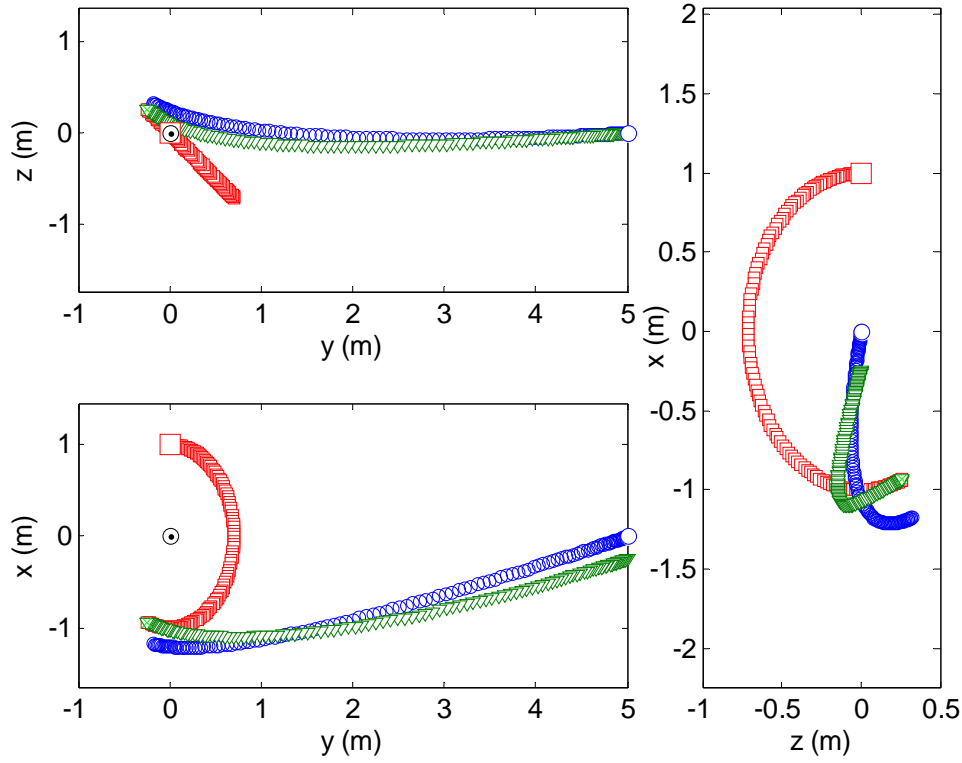


Figure 6. Minimum energy solution (GPOPS): the planar views of the optimal trajectory.

Figure 7 shows a plot of the resulting controls time histories (solid lines) as well as the associated costate time histories (without units) that were used to synthesize the optimal control based on Equation (15) (dashed lines). Figure 8 shows the time history of the difference in position and velocity of the chaser and RSO docking point in the XYZ orbital frame, $\mathbf{e}(t)$, as defined by Equation (12). The time history of the terminal conditions due to transversality, $\zeta(t)$, are shown in Figure 9. Obviously, all the values in Figures 8 and 9 do approach zero as they are supposed to do. Table 2 summarizes the results of optimization in terms of the values of varied parameters, initial value of the costates and Table 3 lists the terminal values of $\mathbf{e}(t)$, $\zeta(t)$ and $H(t)$. Since for numerical solutions the final value of the Hamiltonian does not tell a full story, its complete time history is presented in Figure 10.

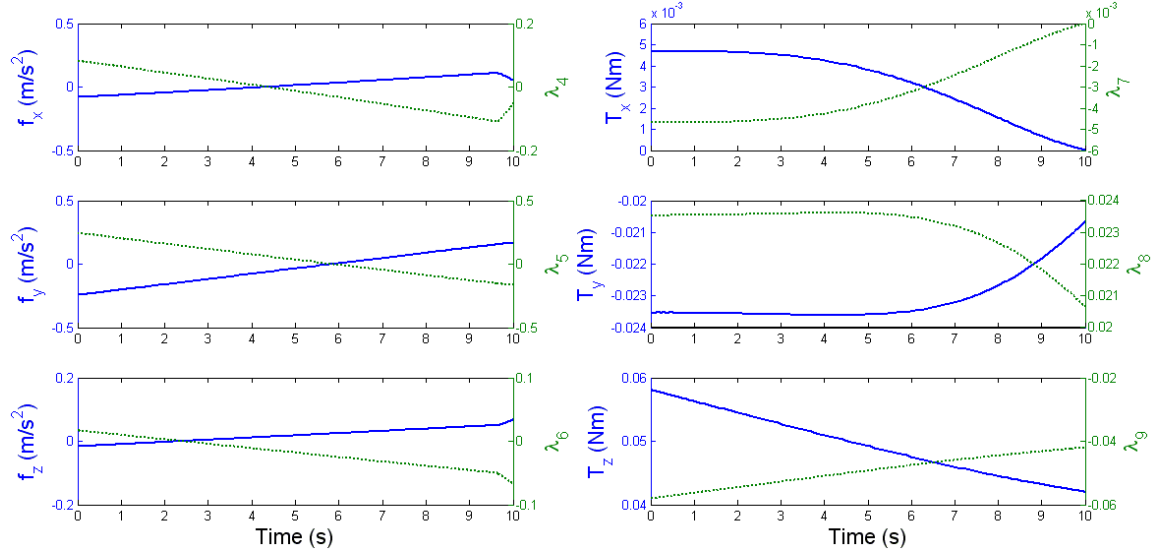


Figure 7. Minimum energy solution (GPOPS): control and associated costate history.

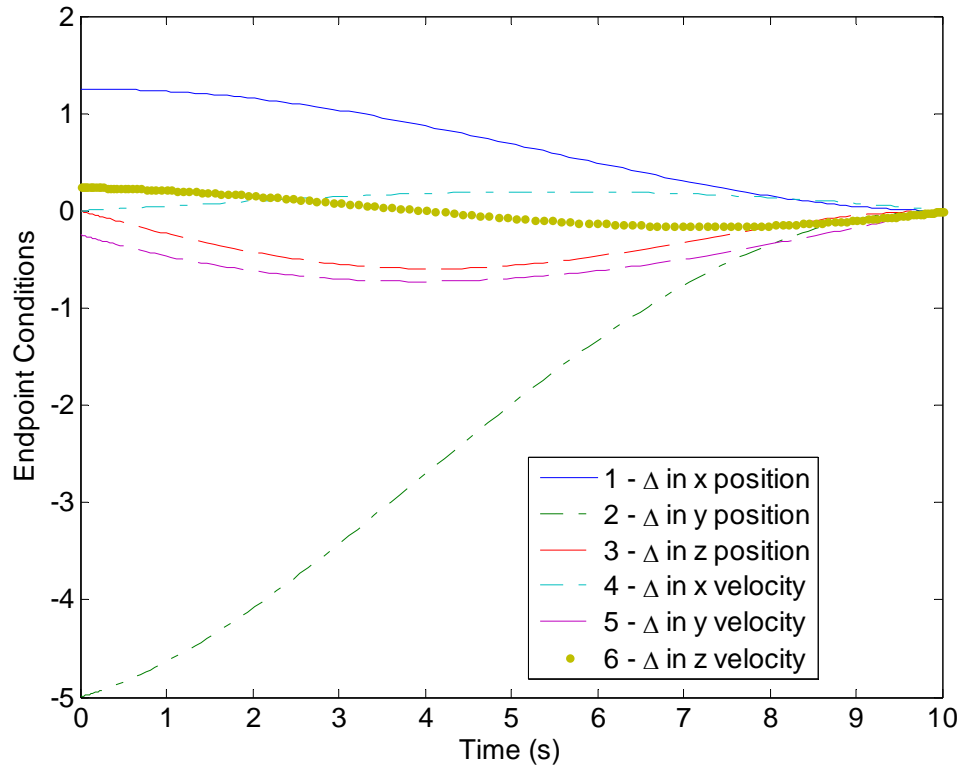


Figure 8. Minimum energy solution (GPOPS): time history of discrepancies in the position and velocity of the chase and RSO docking points.

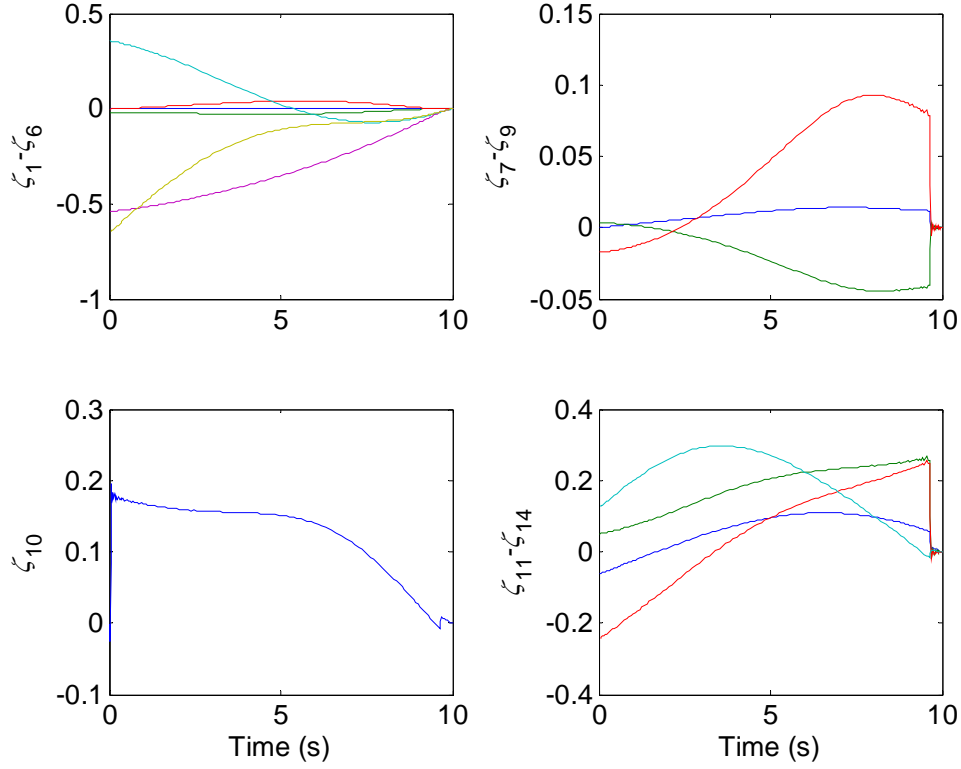


Figure 9. Minimum energy solution (GPOPS): history of the transversality conditions.

Costate	Initial Condition	Costate	Initial Condition	Costate	Initial Condition
λ_1	0.0202713560903818	λ_8	0.0235328656506421	λ_{15}	-0.00357471284206
λ_2	0.0415555942048028	λ_9	-0.058080907113755	λ_{16}	1.060304309630e-10
λ_3	0.00694445082087804	λ_{10}	-0.358237973748141	λ_{17}	-0.0681335012622741
λ_4	0.0816306658067988	λ_{11}	0.556788661928892	λ_{18}	0.0053037488085563
λ_5	0.244702539834469	λ_{12}	0.410549229662363	λ_{19}	0.202974770150912
λ_6	0.0166838774111	λ_{13}	1.41849853483e-05	λ_{20}	-3.422927080667e-10
λ_7	-0.004695462475917	λ_{14}	-2.046214161772e-05	t_f	10

Table 2. The initial values of costates and final time as defined by GPOPS.

endpoint	Resulting Value	endpoint	Resulting Value	endpoint	Resulting Value
$e_1(t_f)$	2.1 e-10	$\zeta_2(t_f)$	-9.9 e-09	$\zeta_9(t_f)$	7.7 e-08
$e_2(t_f)$	-9.4 e-12	$\zeta_3(t_f)$	-1.6 e-09	$\zeta_{10}(t_f)$	5.0 e-08
$e_3(t_f)$	-2.8 e-11	$\zeta_4(t_f)$	1.2 e-11	$\zeta_{11}(t_f)$	1.4 e-07
$e_4(t_f)$	5.1 e-11	$\zeta_5(t_f)$	-4.5 e-09	$\zeta_{12}(t_f)$	6.5 e-08
$e_5(t_f)$	1.7 e-10	$\zeta_6(t_f)$	2.6 e-09	$\zeta_{13}(t_f)$	3.3 e-07
$e_6(t_f)$	1.4 e-10	$\zeta_7(t_f)$	6.4 e-07	$\zeta_{14}(t_f)$	-5.8 e-07
$\zeta_1(t_f)$	3.8 e-07	$\zeta_8(t_f)$	-6.4 e-09	$H(t_f)$	-0.009711

Table 3. Value of terminal and transversality conditions at the final time.

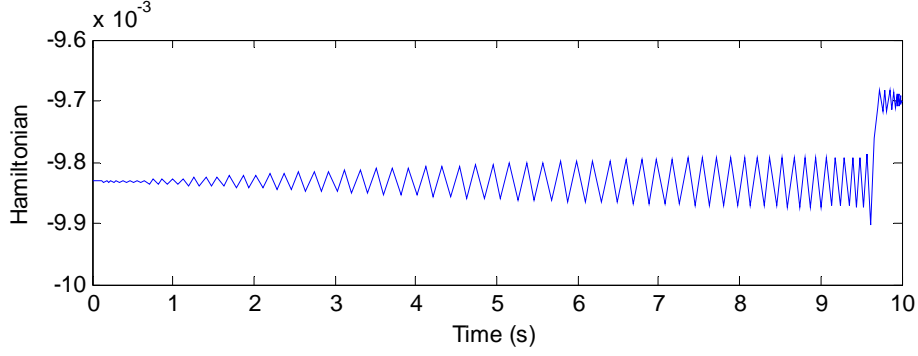


Figure 10. Minimum energy solution (GPOPS): history of the Hamiltonian.

It should be noted that the achieved terminal tolerance of the order of 10^{-7} - 10^{-11} does not necessarily tell about the quality of the solution. The reason for this is that the parameters of the trajectory are only being computed (optimality conditions enforced) at 150 nodes. Another important issue worth mentioning here is that it took 11,869.27 seconds (a little over 3 hours) to produce this 150-node solution of a 10s-long trajectory on a 2.33GHz Dell Precision M90 desktop computer with an Intel T7600 processor and 1Gb of RAM. The initial guess for the solution (required as an input to GPOPS) consisted of two terminal points, one at the initial time and one at the final time. The guess for the initial states corresponded to the initial conditions, while the guess for the final states consisted of zeros for the first 12 states, and the value of $[0,0,0,1]^T$ for the states corresponding to the quaternions. The guess for the control history was 0 at initial and final time for all controls. Less accurate solution, with the lower number of nodes, would obviously require less computational resources (Yakimenko, Xu and Basset 2008). For instance, a 25-node solution, for this case, only required 219.43 seconds (less than 4 minutes) on the same computer. The point is that the solution seems to be feasible and realizable in practice (look at the smooth controls in Figure 7) but can only be used for off-line computations, i.e., in the open-loop guidance schemes. The jump in Hamiltonian value in Figure 10 and other Hamiltonian histories to follow occurs when path constraint from Equation (32) is enforced as in Equation (36).

The next section addresses validation of the GPOPS solution, showing that it is indeed optimal.

2. Verification of the GPOPS Solution with Minimum Principle

As discussed in Chapter II.C.3, the initial guesses provided by the GPOPS solution (Table 2) were used to run an optimization procedure exploiting the optimal controls synthesized using the Minimum Principle. The Quasi-Newton method based optimization routine employing forward shooting and integration of equations of motion using Bogacki-Shampine ordinary differential equation 3 (ODE3) solver with a fixed step of 10^{-3} . This approach results in a solution that has as many as 10,000 points (as opposed to just 150 nodes as in GPOPS). Of course it comes with the cost. Even with the perfect initial guesses for all varied parameters it takes many hours for the optimization process to converge (compared to several minutes with no good initial guess for GPOPS). The MP solution, returned a value of the PI $J=0.11341185$ (compare to $J = 0.1133$ for the GPOPS solution), cost function $\beta=0.001768$, and $P = 1.646 \text{ e-}06$.

As expected, solving the same problem, as in Chapter II.D.1, (a quadratic-control case with path constraints) produces the results, the optimal trajectory, controls and time histories of all parameters, which are pretty close to those produced by GPOPS. The control time histories and those states and costates that were not shown for the GPOCS solution are shown. Specifically, Figure 11 shows the controls time histories, discontinuous at running into a path constraint at $t_d = 9.642s$, Figures 12–14 show the states and corresponding costates of the chaser and target RSO. The values of the initial costates as suggested by MP are shown in Table 4, and the terminal values $\mathbf{e}(t_f)$, $\boldsymbol{\zeta}(t_f)$ and $H(t_f)$ are shown in Table 5. Figure 15 presents the time history of the Hamiltonian.

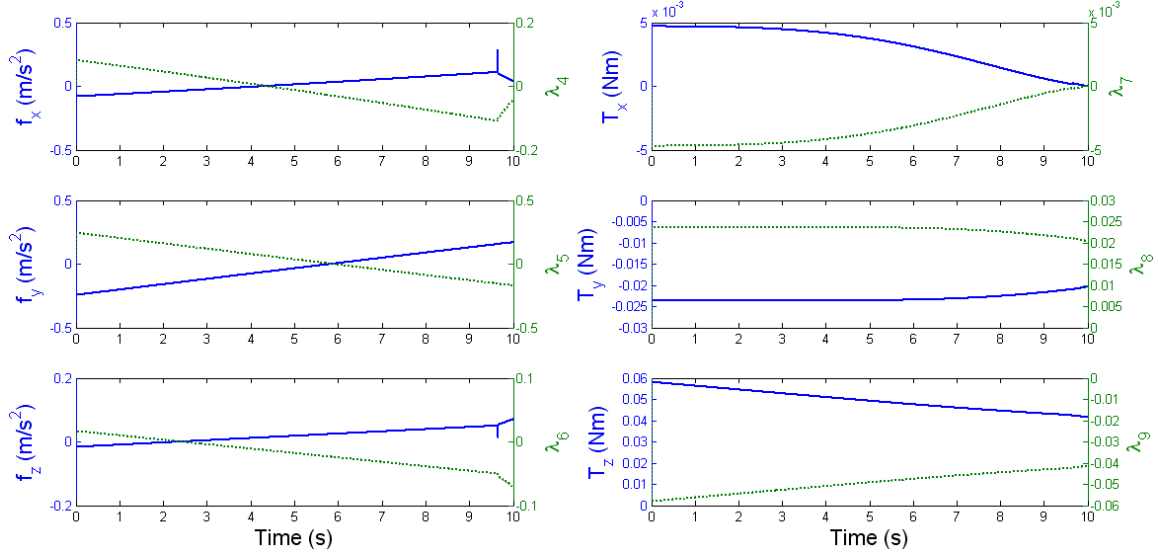


Figure 11. Minimum energy solution (MP): Control time histories and associated costates.

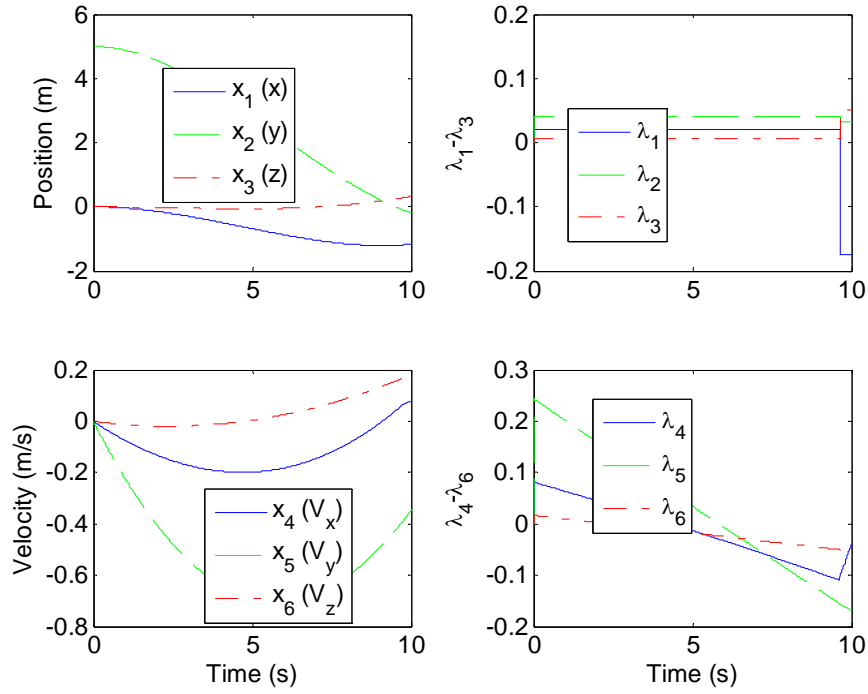


Figure 12. Minimum energy solution (MP): state and costate time-histories for translational variables of the chaser vehicle.

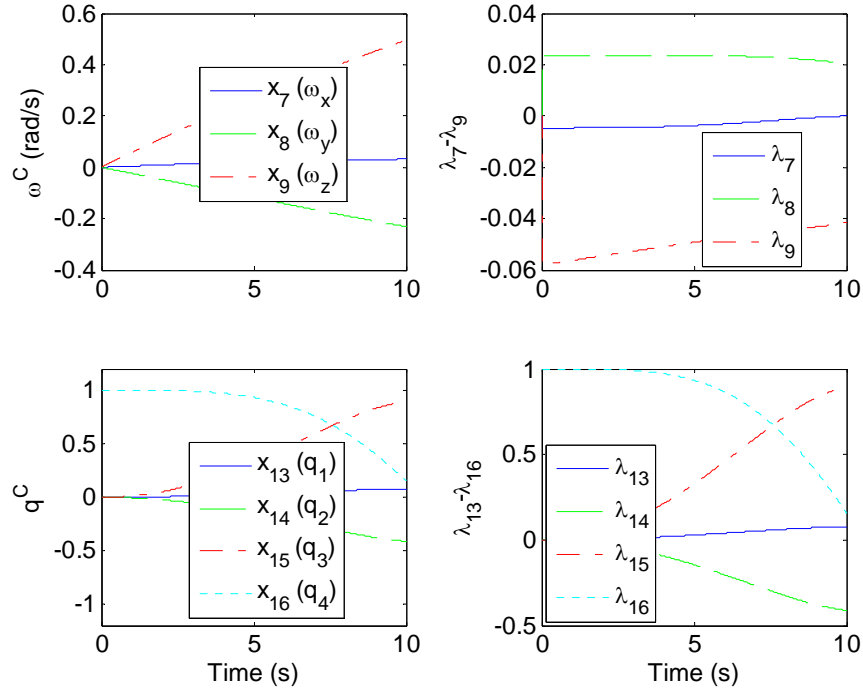


Figure 13. Minimum energy solution (MP): state and costate histories for the defining angular parameters of the chaser vehicle.

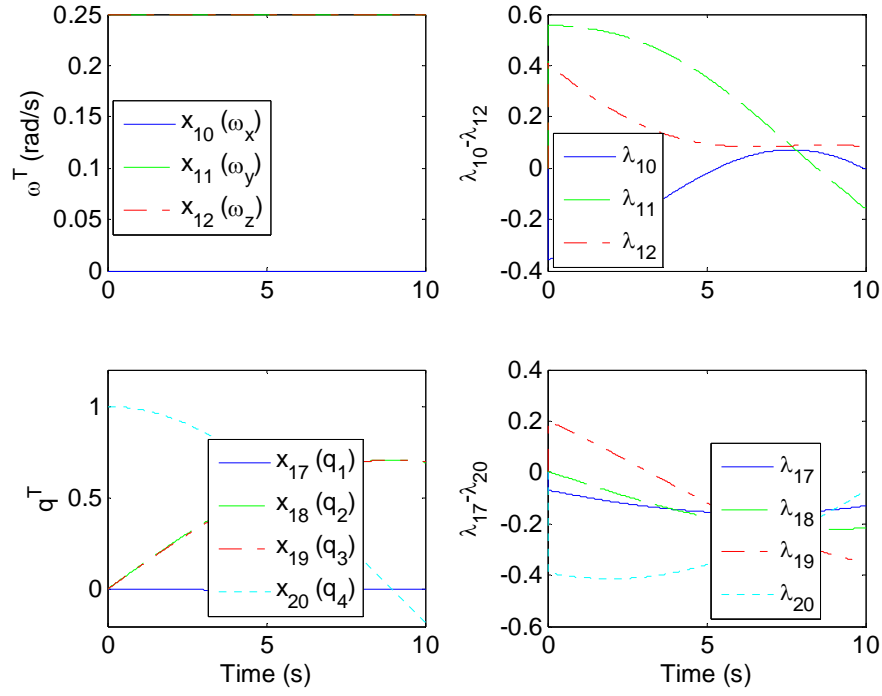


Figure 14. Minimum energy solution (MP): state and costate time histories for the RSO.

<i>Costate</i>	<i>Initial Condition</i>	<i>Costate</i>	<i>Initial Condition</i>	<i>Costate</i>	<i>Initial Condition</i>
λ_1	0.0202713560903818	λ_8	0.0235328656506421	λ_{15}	-0.00357471284206
λ_2	0.0415555942048028	λ_9	-0.058080907113755	λ_{16}	1.060304309630e-10
λ_3	0.00694445082087804	λ_{10}	-0.358237973748141	λ_{17}	-0.0681335012622741
λ_4	0.0816306658067988	λ_{11}	0.556788661928892	λ_{18}	0.0053037488085563
λ_5	0.244702539834469	λ_{12}	0.410549229662363	λ_{19}	0.202974770150912
λ_6	0.0166838774111	λ_{13}	1.41849853483e-05	λ_{20}	-3.422927080667e-10
λ_7	-0.004695462475917	λ_{14}	-2.046214161772e-05	t_f	10

Table 4. The initial values of costates and t_f as defined by MP.

<i>Costate</i>	<i>Resulting Value</i>	<i>Costate</i>	<i>Resulting Value</i>	<i>Costate</i>	<i>Resulting Value</i>
$e_1(t_f)$	0.0001	$\zeta_2(t_f)$	0.0001	$\zeta_9(t_f)$	0.0001
$e_2(t_f)$	-0.0001	$\zeta_3(t_f)$	-0.0001	$\zeta_{10}(t_f)$	-0.0001
$e_3(t_f)$	-0.0006	$\zeta_4(t_f)$	-0.0006	$\zeta_{11}(t_f)$	-0.0006
$e_4(t_f)$	-0.0033	$\zeta_5(t_f)$	-0.0033	$\zeta_{12}(t_f)$	-0.0033
$e_5(t_f)$	0.0012	$\zeta_6(t_f)$	0.0012	$\zeta_{13}(t_f)$	0.0012
$e_6(t_f)$	0.0013	$\zeta_7(t_f)$	0.0013	$\zeta_{14}(t_f)$	0.0013
$\zeta_1(t_f)$	-0.0001	$\zeta_8(t_f)$	-0.0001	$H(t_f)$	-0.0291

Table 5. The value of terminal and transversality conditions at the final time.

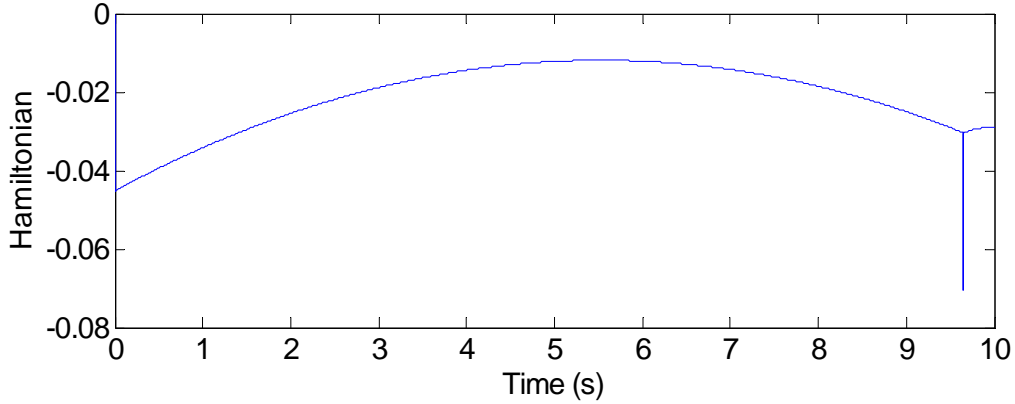


Figure 15. Hamiltonian for the forward shooting minimum-control solution.

Note, some of the numbers in Table 4 appear in the bold font. These, are the only numbers that are different from the GPOPS solution (Table 4). Therefore, it took hours and hours to correct initial values for just a few costates, primarily four of them, λ_{13} , λ_{14} , λ_{16} , and λ_{20} . Obviously, it is due to the fact that because if integration the solution is quite sensitive to the initial values of varied parameters. Also, the accuracy of GPOPS solution

(Table 3) cannot possibly be matched. However, the point of the entire exercise was to prove that GPOPS does provide a solution that is very close to the truly optimal one, and it was proven.

E. OBTAINING AND ANALYZING THE MINIMUM TIME SOLUTION

1. Minimum Time Solution with GPOPS

The 3D trajectory and 2D projections of the trajectories as a result of the GPOPS solution are shown in Figures 16 and 17, respectively. As seen, they are quite different from those shown in Figures 5 and 6. Also notice that, although the positions and velocities of the angular velocities match, the vehicle orientations do not. The resulting control history is shown in Figure 18. The f_z control (of the translational motion in the z direction) turns out to be highly oscillating. The endpoint conditions, taken from the state constraints on the endpoint as well as the transversality conditions are shown in Figure 19 and 20. The values of the initial costates as suggested by GPOPS are shown in Table 6 and the terminal conditions of the boundary equations are shown in Table 7. The time history for the Hamiltonian is presented in Figure 21.

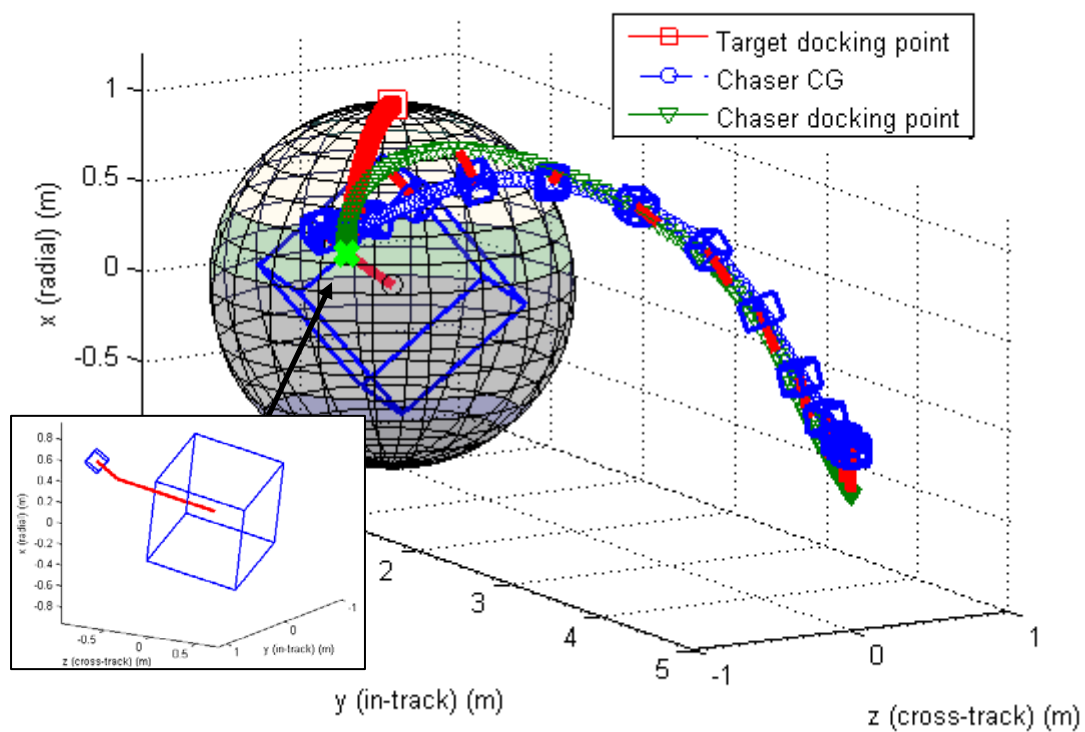


Figure 16. Minimum time solution (GPOPS): The 3D view of the optimal trajectory. A close-up of the final position is shown in the exploded view.

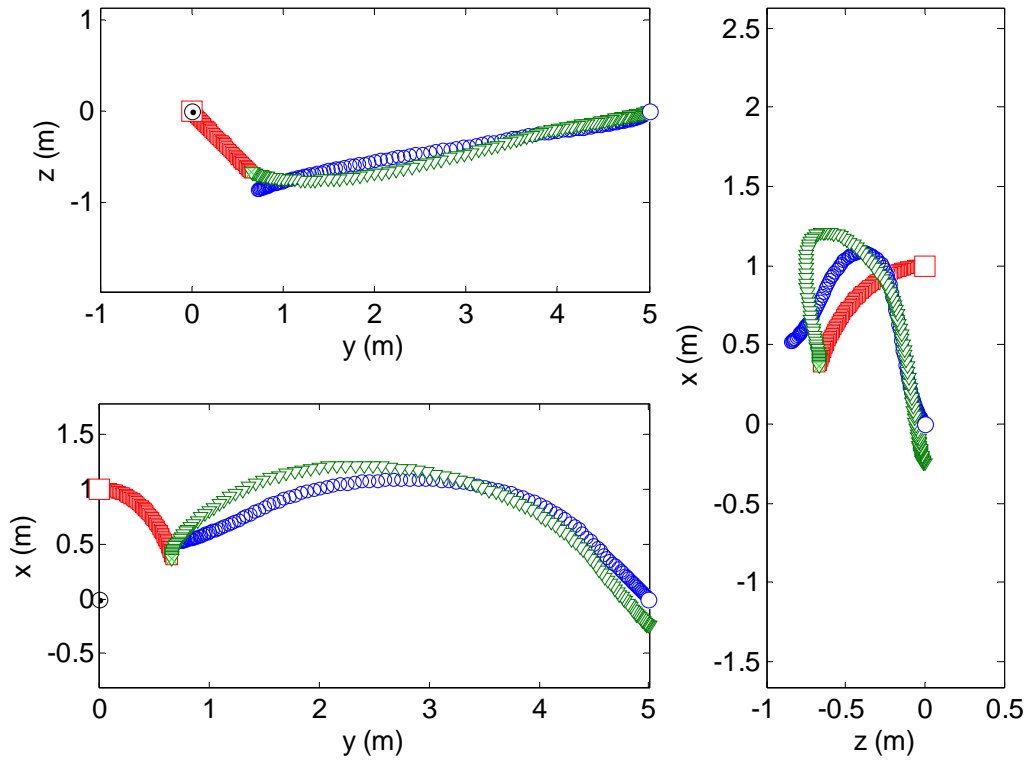


Figure 17. Minimum time solution (GPOPS): the 2D plots of optimal rendezvous trajectory.

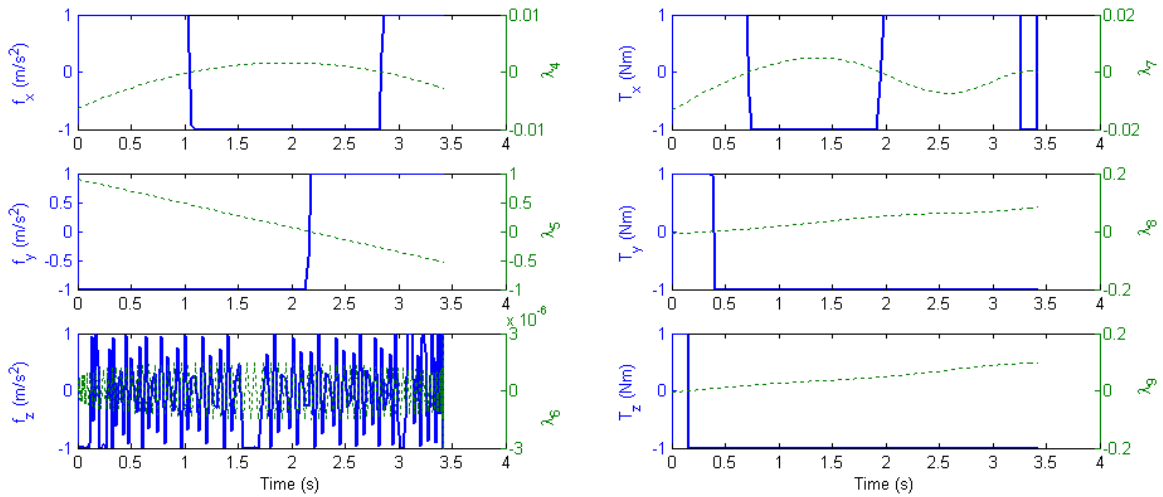


Figure 18. Minimum time solution (GPOPS): the control and associated costate history.

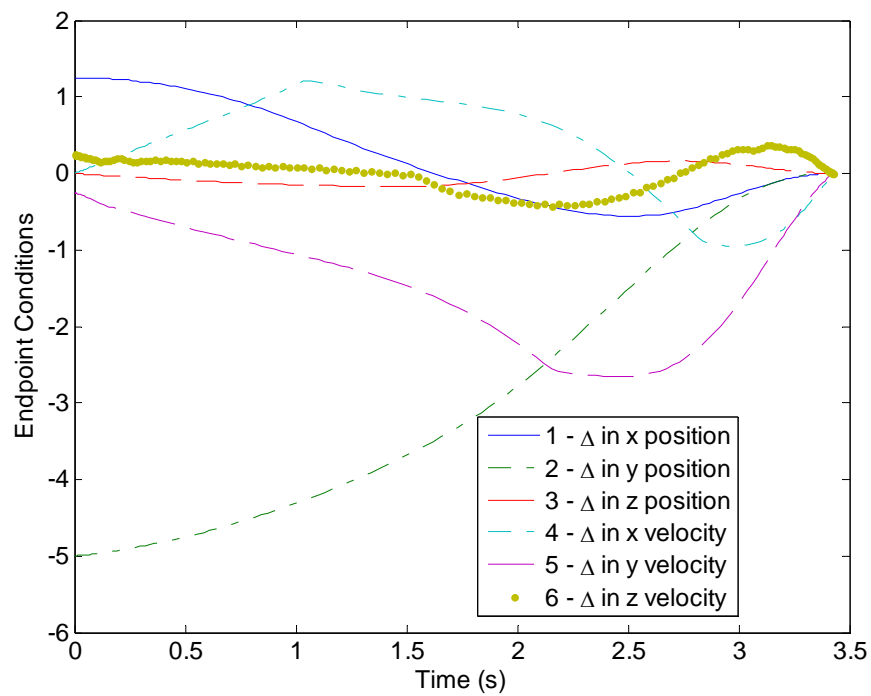


Figure 19. Minimum time solution (GPOPS): time history of discrepancies in the position and velocity of the chaser and RSO docking point.

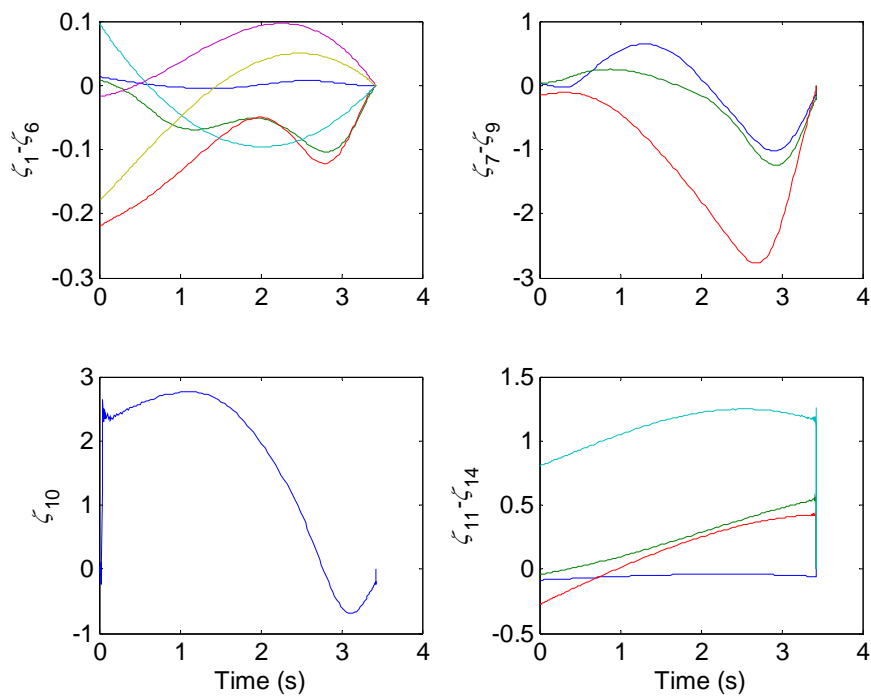


Figure 20. Minimum time solution (GPOPS): history of transversality conditions.

λ_i	Initial Condition	λ_i	Initial Condition	λ_i	Initial Condition
λ_1	0.000834092040702561	λ_8	-0.00781391345181247	λ_{15}	-0.0670307214682799
λ_2	0.418560914087817	λ_9	-0.00529954719645793	λ_{16}	-2.47014407229089
λ_3	7.46232139969827e-08	λ_{10}	-0.0966015054618492	λ_{17}	-0.365359323439127
λ_4	-0.00634589252204042	λ_{11}	0.017598938378927	λ_{18}	0.0400467496481145
λ_5	0.901334242652277	λ_{12}	-0.719562963054045	λ_{19}	-0.562241844247041
λ_6	1.04649495108653e-07	λ_{13}	-0.0420682846405162	λ_{20}	-1.6978671241818
λ_7	-0.0133948065189441	λ_{14}	-0.0394802339637517	t_f	3.4237

Table 6. The initial values of costates and as defined by GPOPS.

endpoints	Resulting Value	endpoints	Resulting Value	endpoints	Resulting Value
$e_1(t_f)$	-3.1 e-08	$\zeta_2(t_f)$	3.3 e-07	$\zeta_9(t_f)$	-1.9 e-06
$e_2(t_f)$	-3.8 e-08	$\zeta_3(t_f)$	-6.9 e-08	$\zeta_{10}(t_f)$	3.8 e-06
$e_3(t_f)$	4.9 e-08	$\zeta_4(t_f)$	-5.2 e-14	$\zeta_{11}(t_f)$	3.2 e-07
$e_4(t_f)$	-1.8 e-07	$\zeta_5(t_f)$	1.0 e-07	$\zeta_{12}(t_f)$	-5.1 e-07
$e_5(t_f)$	5.1 e-07	$\zeta_6(t_f)$	1.0 e-07	$\zeta_{13}(t_f)$	-2.0 e-07
$e_6(t_f)$	-1.7 e-08	$\zeta_7(t_f)$	-3.5 e-07	$\zeta_{14}(t_f)$	1.1 e-07
$\zeta_1(t_f)$	-3.3 e-10	$\zeta_8(t_f)$	9.7 e-06	$H(t_f)$	-0.2801

Table 7. Value of terminal conditions at the final time as calculated by GPOPS.

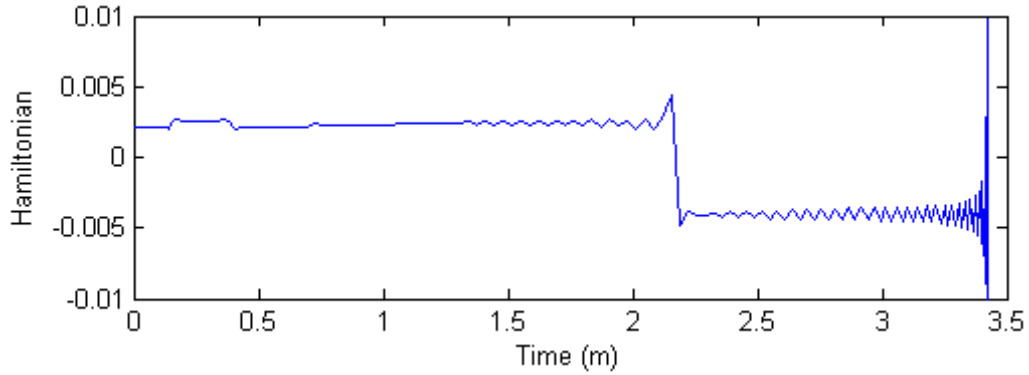


Figure 21. Minimum time solution (GPOPS): history of the Hamiltonian.

Closer inspection of the GPOPS solution for λ_6 in Figure 18 (it jumps back and forth? around zero value) suggests a presence of a singular control. This is indirectly confirmed by oscillations of the Hamiltonian as well (Yakimenko, Xu and Basset 2008). Hence, the f_z control is not optimal and infeasible.

With 150 nodes, the computational time to arrive at the solution shown above (3.4237 second maneuver) was 8,929.55 seconds (~2.5 hours). The initial guess for the solution was the same as the one used for the minimum energy case. For comparison, with 25 nodes the required computational time can be brought down to 100.77 seconds

(<2 min). However, as seen from Figure 22, the controls in this case are even less optimal than in Figure 18, and of course the Hamiltonian (Figure 23) looks worse than that of Figure 21.

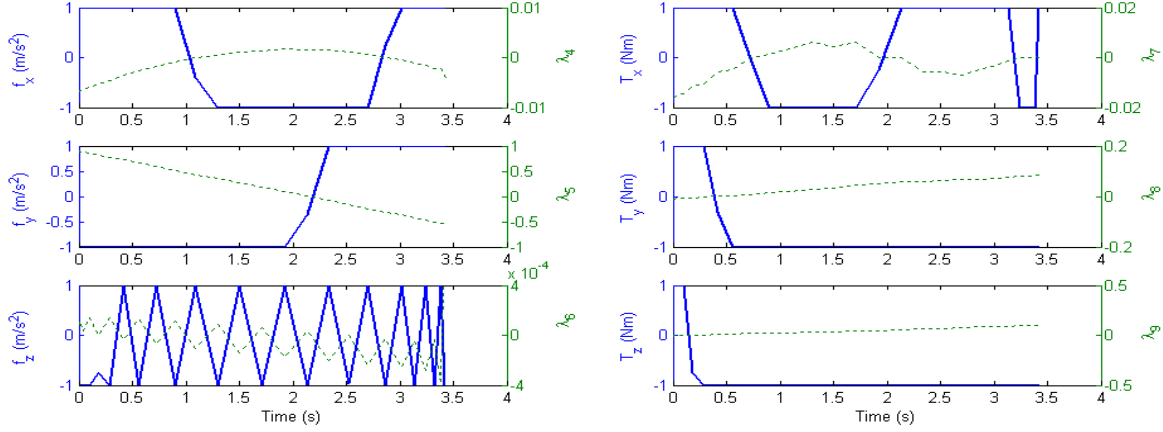


Figure 22. Minimum time solution (GPOPS): The associated costate history resulting from a 25 node solution.

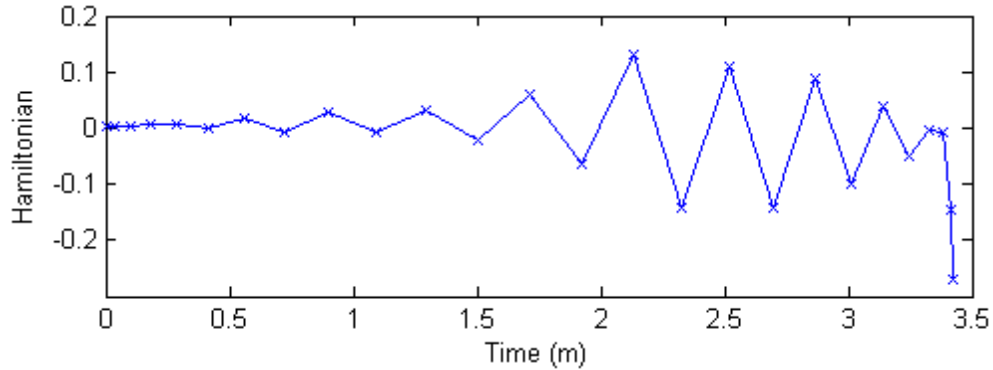


Figure 23. Minimum time solution (GPOPS): Hamiltonian for the associated 25-node solution.

2. Verification of the GPOPS Solution with Minimum Principle

The minimum time rendezvous problem is approached in a similar fashion of the minimum-control one. First, the problem is investigated using a β based on Equation (52). A major difference that arises compared to the minimum-control solution is the existence of a singular control in u_3 , which controls acceleration in the z orbital direction.

As discussed in Chapter II.B.3, a singular control exists and needs special consideration for numerical implementation. The optimal control structure was assumed to have the following form:

$$u_3 = \begin{cases} -1, & 0 \leq t < t_1 \\ 0, & t_1 \leq t < t_2 \\ 1, & t_2 \leq t < t_3 \\ 0, & t_3 \leq t < t_4 \\ -1, & t_4 \leq t < t_5 \\ 0, & t_5 \leq t < t_6 \\ 1, & t_6 \leq t < t_f \end{cases} \quad (53)$$

Accordingly, the vector of varied parameters \mathbf{X} was augmented with the switching times t_1, \dots, t_6 :

$$\mathbf{X} = [\lambda_1(t_f), \dots, \lambda_{20}(t_f), \lambda_1(t_d), \dots, \lambda_{20}(t_d), t_1, \dots, t_6, t_f]' \quad (54)$$

Having instances t_1, \dots, t_6 as varied parameters implied that the search was made among multiple control profiles including traditional bang-bang control, like $u_{3\min} - u_{3\max}$ (when $t_1 = t_2$ and $t_i = t_f$, $i = 3, \dots, 6$), or $u_{3\max} - u_{3\min}$ (when $t_1 = t_2 = 0$, $t_3 = t_4$ and $t_5 = t_6 = t_f$, $i = 3, \dots, 6$) allows calculation of a control history even in the presence of a singular arc.

The resulting trajectory is shown in Figures 24 and 25. The corresponding optimal controls profiles are presented in Figure 26. As seen, it was $u_{3\min} - 0 - u_{3\max} - u_{3\min}$ profile that was found to be optimal for the u_3 control (i.e., $t_5 = t_6 = t_f$). The time histories for the remaining controls, u_1, u_2, u_4, u_5, u_6 , match those for the GPOPS solution (Figure 15). Figures 27-29 show the state costates time histories for the chaser and target RSO, respectively. The values of the initial costates as suggested by MP are shown in Table 8

and the error in satisfying boundary conditions are shown in Table 9 (the cost $\beta = 0.002$, $P=0.00124$ Figure 30 presents the Hamiltonian (compare it with that of Figure 21 and 22)).

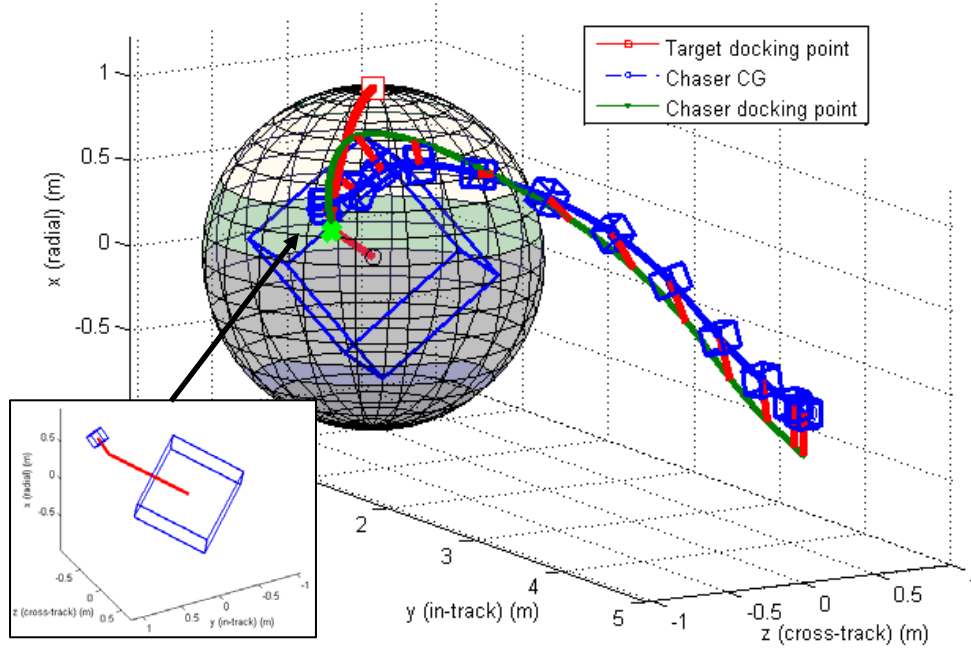


Figure 24. Minimum time solution (MP): 3D optimal rendezvous trajectory.

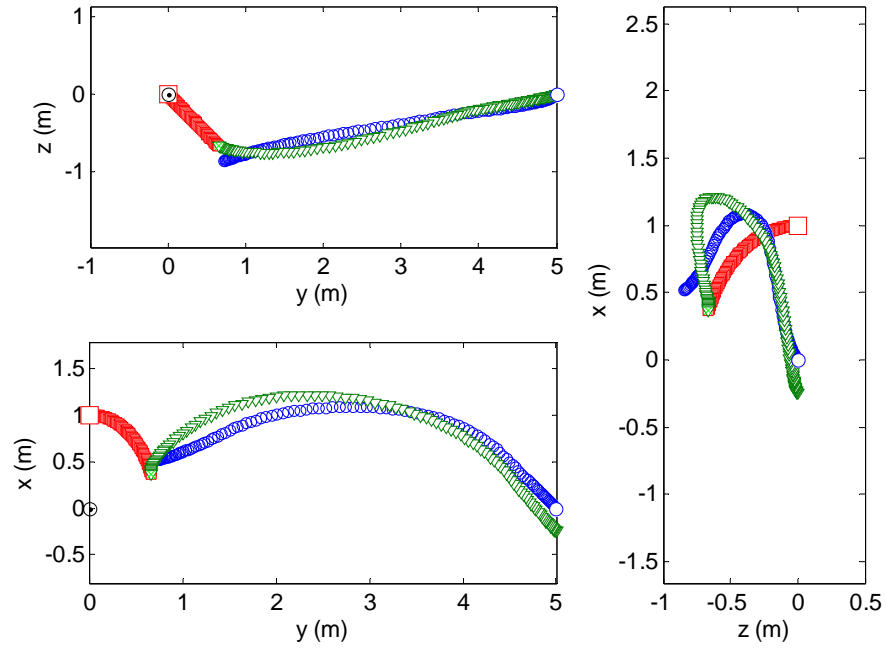


Figure 25. Minimum time solution (MP): The 2D projections of the optimal rendezvous trajectory.

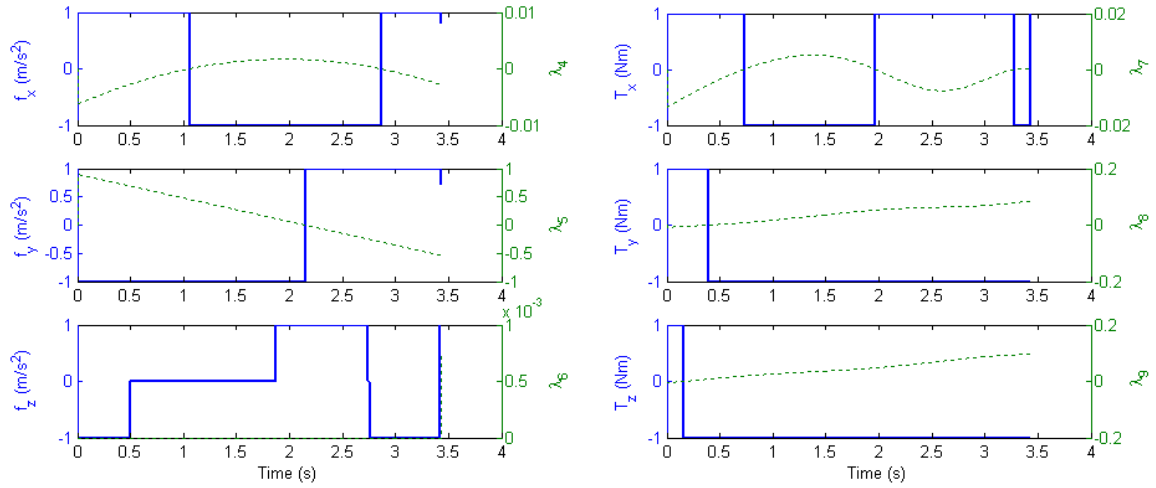


Figure 26. Minimum time solution (MP): control histories and associated switching conditions.

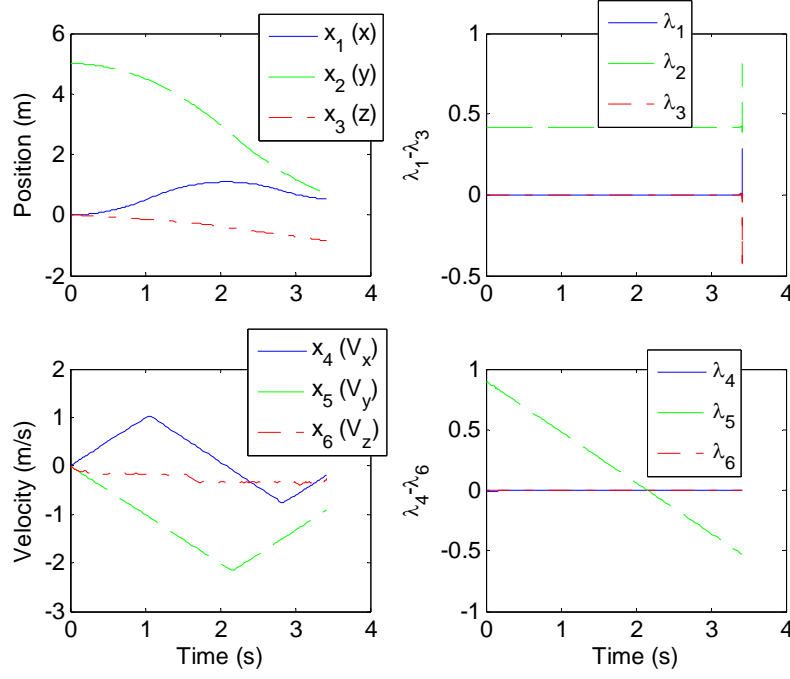


Figure 27. Minimum time solution (MP): State and costate histories for the translational variables of the chaser vehicle.

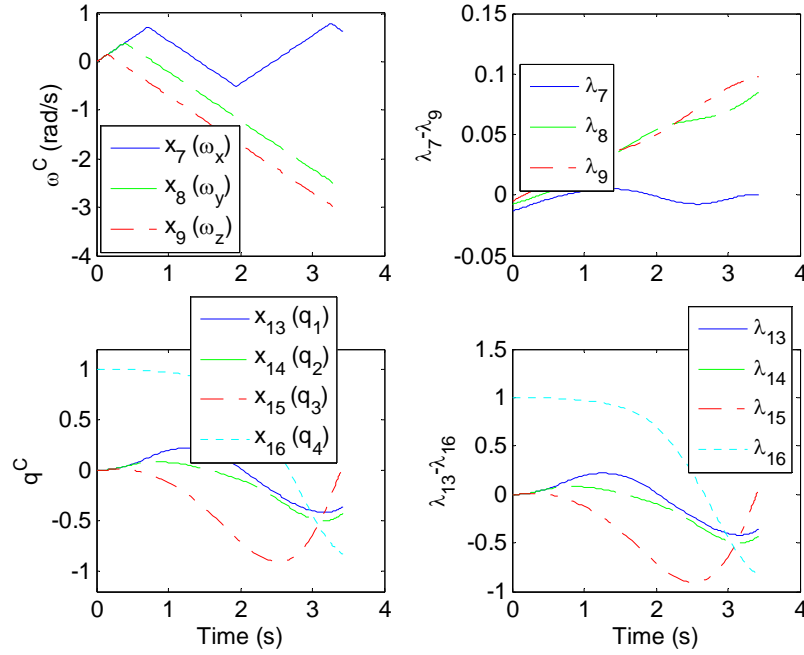


Figure 28. Minimum time solution (MP): state and costate histories for the defining angular parameters of the chaser vehicle.

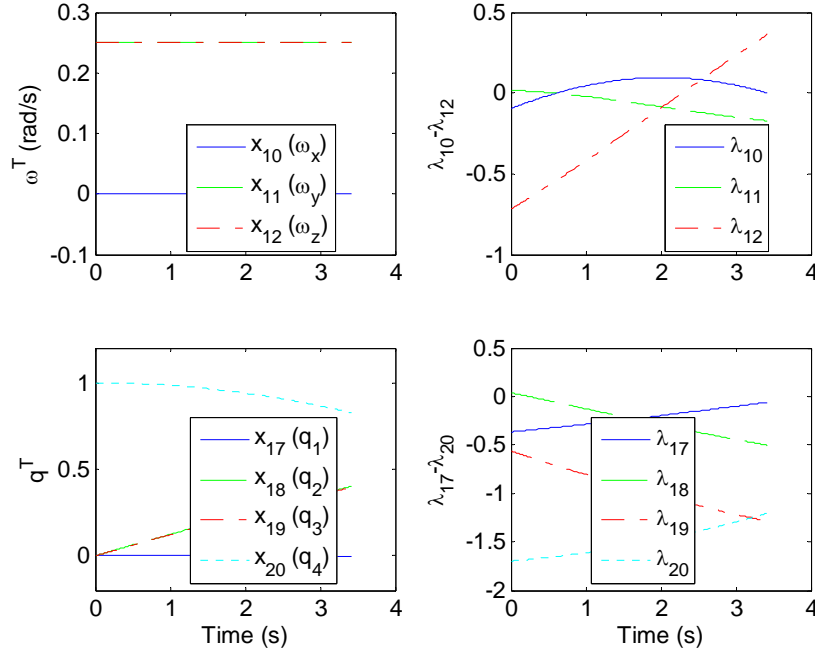


Figure 29. Minimum time solution (MP): state and costate time histories for the RSO.

λ_i	Initial Condition	λ_i	Initial Condition	λ_i	Initial Condition
λ_1	0.000813270815835508	λ_8	-0.0079043944784984	λ_{15}	-0.067030975848676
λ_2	0.41856021740054	λ_9	-0.0052514744594656	λ_{16}	-2.47013927846309
λ_3	0	λ_{10}	-0.0966015055875181	λ_{17}	-0.365359450504476
λ_4	-0.00632550328119496	λ_{11}	0.017598938703864	λ_{18}	0.0400468901069352
λ_5	0.90134410424794	λ_{12}	-0.719562962786195	λ_{19}	-0.562241768137901
λ_6	0	λ_{13}	-0.042050296747438	λ_{20}	-1.6978671241818
λ_7	-0.0135549835134852	λ_{14}	-0.0395088309550527	t_f	3.432

Table 8. The initial values of costates and t_f as defined by MP.

e_i	Resulting Value	e_i	Resulting Value	e_i	Resulting Value
$e_1(t_f)$	0.016	$\zeta_2(t_f)$	0.0028	$\zeta_9(t_f)$	0.0096
$e_2(t_f)$	-0.014	$\zeta_3(t_f)$	-0.0018	$\zeta_{10}(t_f)$	0.0061
$e_3(t_f)$	0.0016	$\zeta_4(t_f)$	0.00013	$\zeta_{11}(t_f)$	0.00048
$e_4(t_f)$	-0.017	$\zeta_5(t_f)$	-0.00059	$\zeta_{12}(t_f)$	-0.00079
$e_5(t_f)$	0.029	$\zeta_6(t_f)$	-0.00077	$\zeta_{13}(t_f)$	0.0010
$e_6(t_f)$	0.0080	$\zeta_7(t_f)$	0.0066	$\zeta_{14}(t_f)$	6.6 e-05
$\zeta_1(t_f)$	1.9 e-05	$\zeta_8(t_f)$	-0.0063	$H(t_f)$	0.15

Table 9. Value of terminal and transversality conditions at the final time.

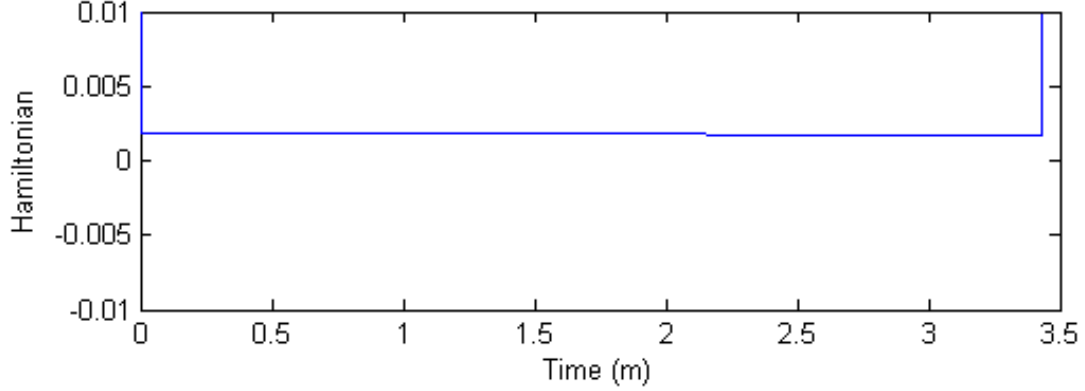


Figure 30. Minimum time solution (MP): Hamiltonian for the forward shooting minimum time solution.

Again, the bold numbers in Table 8 indicate the differences as compared to GPOPS quasi-optimal solution. As opposed to the minimum-control case, when truly optimal solution happened to have almost the same values of the initial costates, in the minimum time case implying a singular control, the optimal solution involved more variations from the solution provided by GPOPS. Of course, as in the minimum-control case presented in Chapter II.D.2, the indirect-method-based optimization was run recursively multiple times for several days in order to converge. Nevertheless, the errors in satisfying the terminal conditions (Table 9) were of several orders higher than that of claimed by the GPOPS solution (Table 7). The next section provides some more details on this issue.

3. Propagation of the GPOPS Solution

Once again, for the solutions provided by GPOPS, vehicles dynamics are only satisfied in a limited number of nodes. That is why the error in meeting all constraints is so negligibly small as compared to solutions provided by the methods involving integration of equations of motion (shooting). However, if we integrate the GPOPS controls, we will end up with about the same (lower) accuracy as the shooting approach.

To illustrate it, let us, for example, take the controls for the minimum time solution (including a “weird” control for f_z) and integrate it. The result of integrating the equations of motion derived in Chapters II.A and control switches in Chapter II.B with a fixed time step of 0.0001 seconds is shown in Figures 31 and 32 (the zero-order hold of

the last control inputs was used). As seen, the trajectories are practically the same as in Figures 16 and 17, but the endpoint discrepancies, summarized in Table 10, obviously grew up. The endpoint conditions of the state variables are shown in Table 10. Note, that the endpoint conditions of the transversality conditions are not known because the costates are not propagated.

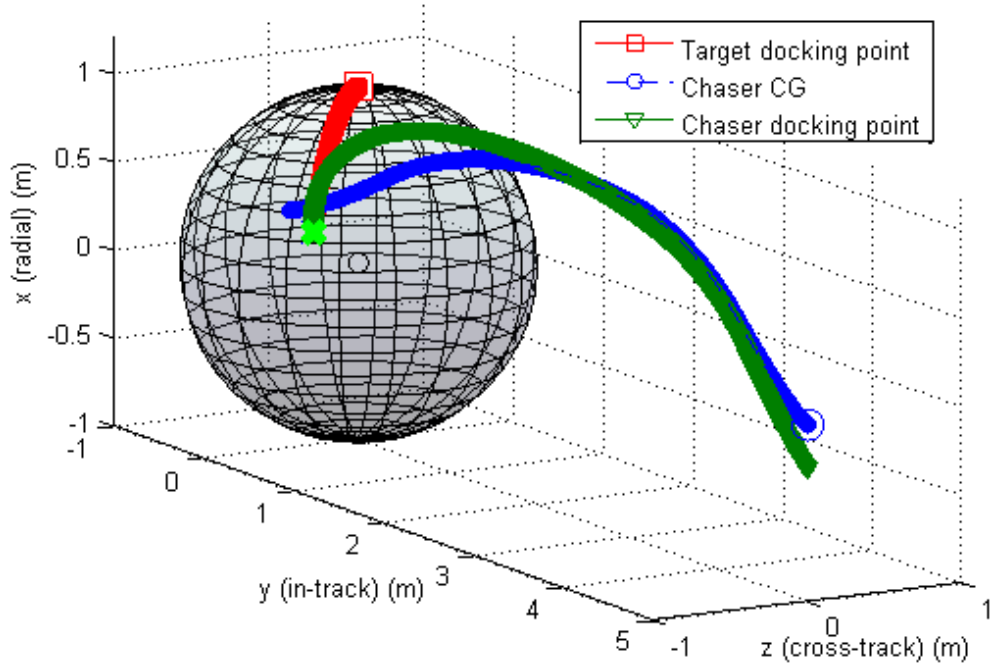


Figure 31. Propagated trajectory using the minimum time control history from GPOPS.

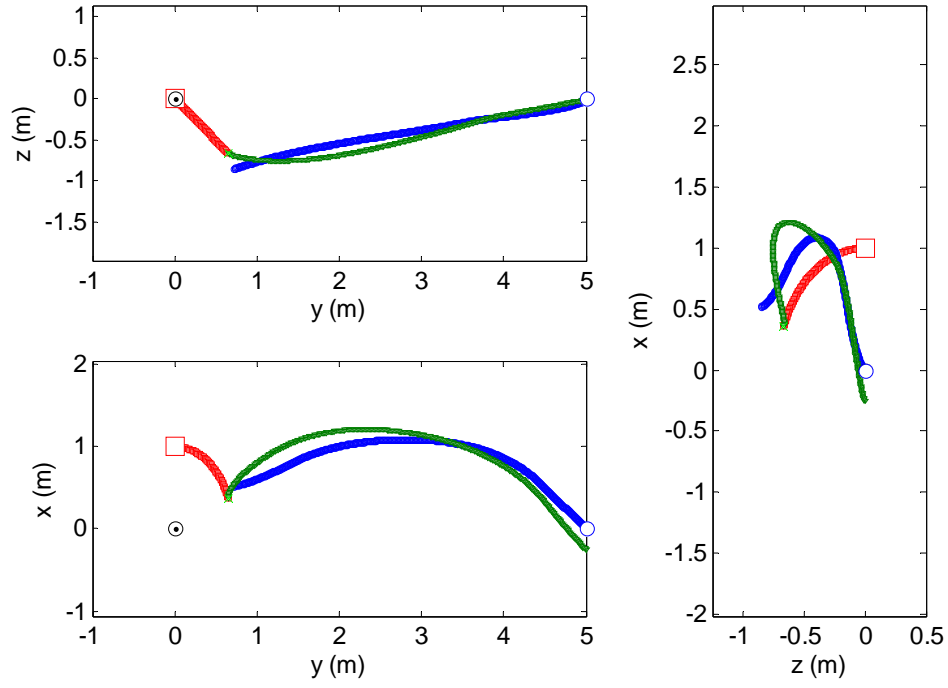


Figure 32. Propagated trajectory using the minimum time control history from GPOPS.

e	<i>Resulting Value</i>	e	<i>Resulting Value</i>	e	<i>Resulting Value</i>
$e_1(t_f)$	-0.00027	$\zeta_2(t_f)$	N/A	$\zeta_9(t_f)$	N/A
$e_2(t_f)$	-9.3e-05	$\zeta_3(t_f)$	N/A	$\zeta_{10}(t_f)$	N/A
$e_3(t_f)$	0.00037	$\zeta_4(t_f)$	N/A	$\zeta_{11}(t_f)$	N/A
$e_4(t_f)$	-0.00092	$\zeta_5(t_f)$	N/A	$\zeta_{12}(t_f)$	N/A
$e_5(t_f)$	-0.00035	$\zeta_6(t_f)$	N/A	$\zeta_{13}(t_f)$	N/A
$e_6(t_f)$	0.00026	$\zeta_7(t_f)$	N/A	$\zeta_{14}(t_f)$	N/A
$\zeta_1(t_f)$	N/A	$\zeta_8(t_f)$	N/A	$H(t_f)$	N/A

Table 10. Value of terminal conditions at the final time.

III. OPTIMIZATION OF A SPACECRAFT MANEUVER FOR CLOSE APPROACH AND DOCKING WITH A TUMBLING OBJECT

This chapter takes the study further by considering translational actuators fixed in the chaser spacecraft body frame (better representing an actual system), adding the requirement to match attitude and angular velocity of the chaser and RSO in preparation for a docking procedure (along with the requirement of matching position and velocity of chaser and RSO docking points), and considering one more PI (minimum fuel). The optimal control problem posed in Chapter II is, therefore, reformulated to take into consideration the effects on dynamics and the extra constraints.

A. RENDEZVOUS MODELING AND OPTIMIZATION PROBLEM FORMULATION

Using these controls, we would like to bring the two spacecraft from some initial conditions, given by 20 initial values of states $x_i(t_0)$, $i = 1, \dots, 20$, to a docking-enabling condition described by matching the chaser's and RSO's docking station final positions and velocity vectors as well as matching the orientation and angular velocity of the two vehicles. Defining ${}^O_T \mathbf{R}$ as the rotation matrix to convert from the body frame of the target to the orbital frame, we can express the desired terminal conditions. These conditions of matching spacecraft docking positions and velocities, in the matrix form, are identical to the conditions shown in Equation (12). Matching orientation and angular rates leads to seven more boundary equations at t_f :

$$\begin{bmatrix} q_1^T \\ q_2^T \\ q_3^T \\ q_4^T \end{bmatrix} - \begin{bmatrix} q_1^C \\ q_2^C \\ q_3^C \\ q_4^C \end{bmatrix} = \begin{bmatrix} e_7 \\ e_8 \\ e_9 \\ e_{10} \end{bmatrix} \quad (55)$$

and

$$\begin{bmatrix} \omega_1^T \\ \omega_2^T \\ \omega_3^T \end{bmatrix} - \begin{bmatrix} \omega_1^C \\ \omega_2^C \\ \omega_3^C \end{bmatrix} = \begin{bmatrix} e_{11} \\ e_{12} \\ e_{13} \end{bmatrix} \quad (56)$$

If Equations (55) and (56) are satisfied, that would provide: ${}^o_c \mathbf{R} = {}^o_T \mathbf{R}$ and ${}^o \boldsymbol{\omega}^C = {}^o \boldsymbol{\omega}^T$.

Therefore, the conditions in Equation (12) can be rewritten as:

$${}^o_T \mathbf{R} \begin{bmatrix} d_1^T - d_1^C \\ d_2^T - d_2^C \\ d_3^T - d_3^C \end{bmatrix} - \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} \quad (57)$$

and

$${}^o \boldsymbol{\omega}^T \times {}^o_T \mathbf{R} \begin{bmatrix} d_1^T - d_1^C \\ d_2^T - d_2^C \\ d_3^T - d_3^C \end{bmatrix} - \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} e_4 \\ e_5 \\ e_6 \end{bmatrix} \quad (58)$$

The resulting complete set of desired final conditions are $[e_1(t_f), \dots, e_{13}(t_f)]^T = \mathbf{0}$.

For optimal control problem in this chapter, three different performance indices each of the (Mayer) form

$$J = \int_{t_0}^{t_f} f_0 dt \quad (59)$$

with the running cost

$$f_0 = 1 \quad (60)$$

for minimum time,

$$f_0 = \frac{1}{2} (u_1^2 + u_2^2 + u_3^2 + u_4^2 + u_5^2 + u_6^2) \quad (61)$$

for minimum quadratic-control expenditure (i.e., minimum energy), and

$$f_0 = (|u_1| + |u_2| + |u_3| + |u_4| + |u_5| + |u_6|) \quad (62)$$

for minimum-control are considered. Each case has a maximum finite value chosen for the final time t_f but is still considered a free (varied) parameter. The next step is to convert this optimal control problem to a TPBVP using MP (Pontryagin et al. 1964; Bryson and Ho 1975)

B. MINIMUM PRINCIPLE FORMULATION

This section deals with the MP formulation and synthesis of the optimal control. Using the same state vector of the complete system as shown in Equation (10), (the differential equations for parameters of chaser and target quaternions are standard and can be found in Chapter II), and the control vector shown in Equation (11), we write down the Hamiltonian:

$$H(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{u}) := f_0 + (\boldsymbol{\lambda}, \dot{\mathbf{x}}) + \mu h \quad (63)$$

where operator (...) on the right-hand side denotes a scalar product of two vectors, and $\boldsymbol{\lambda} \in \mathbb{R}^{N_x}$ is a costate vector where its differential equations are to be defined later in this section. The value of μ is a constant and is dictated by the same constraint as in Equation (36) when considering path constraints.

Again, as in Chapter II.B.2, this constraint is that the CM of the chaser spacecraft must remain at a distance larger than some minimum distance (a “keep out” sphere with a radius r) from the CM of the RSO, which is coincident with the origin of the orbit frame. This assures that the chaser vehicle will not pass through the target vehicle in order to reach the docking position shown mathematically in Equation (32).

The entire Hamiltonian is expressed, with respect to the state vector \mathbf{x} defined by Equation (10), in Equation (64).

$$\begin{aligned}
H(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{u}) = & 1 + \lambda_1 x_4 + \lambda_2 x_5 + \lambda_3 x_6 \\
& + \frac{1}{m} \left(\begin{aligned} & \lambda_4 (2\Omega x_5 + 3\Omega^2 x_1 + (x_{16}^2 + x_{13}^2 - x_{14}^2 - x_{15}^2) u_1 + 2(x_{13}x_{14} - x_{15}x_{16}) u_2 + 2(x_{13}x_{15} + x_{14}x_{16}) u_3) \\ & + \lambda_5 (-2\Omega x_4 + 2(x_{13}x_{14}^2 + x_{15}x_{16}) u_1 + (x_{16}^2 - x_{13}^2 + x_{14}^2 - x_{15}^2) u_2 + 2(x_{14}x_{15} - x_{13}x_{16}) u_3) \\ & + \lambda_6 (-\Omega^2 x_3 + 2(x_{13}x_{15} - x_{14}x_{16}) u_1 + 2(x_{14}x_{15} + x_{13}x_{16}) u_2 + (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2) u_3) \end{aligned} \right) \\
& + \frac{I_{22}^C - I_{33}^C}{I_{11}^C} (x_8 x_9 + u_4) \lambda_7 + \frac{I_{33}^C - I_{11}^C}{I_{22}^C} (x_7 x_9 + u_5) \lambda_8 + \frac{I_{11}^C - I_{22}^C}{I_{33}^C} (x_8 x_7 + u_6) \lambda_9 \\
& + \frac{I_{22}^T - I_{33}^T}{I_{11}^T} (x_{11} x_{12}) \lambda_{10} + \frac{I_{33}^T - I_{11}^T}{I_{22}^T} (x_{10} x_{12}) \lambda_{11} + \frac{I_{11}^T - I_{22}^T}{I_{33}^T} (x_{11} x_{10}) \lambda_{12} \\
& + \lambda_{13} \frac{1}{2} \left(\begin{aligned} & \left(x_9 - ((x_{16})^2 - (x_{13})^2 - (x_{14})^2 + (x_{15})^2) \Omega \right) x_{14} \\ & - (x_8 - 2(x_{14}x_{15} - x_{13}x_{16}) \Omega) x_{15} + (x_7 - 2(x_{13}x_{15} + x_{14}x_{16}) \Omega) x_{16} \end{aligned} \right) \\
& + \lambda_{14} \frac{1}{2} \left(\begin{aligned} & - (x_9 - ((x_{16})^2 - (x_{13})^2 - (x_{14})^2 + (x_{15})^2) \Omega) x_{13} \\ & + (x_8 - 2(x_{14}x_{15} - x_{13}x_{16}) \Omega) x_{16} + (x_7 - 2(x_{13}x_{15} + x_{14}x_{16}) \Omega) x_{15} \end{aligned} \right) \\
& + \lambda_{15} \frac{1}{2} \left(\begin{aligned} & - (x_7 - 2(x_{13}x_{15} + x_{14}x_{16}) \Omega) x_{14} \\ & + (x_8 - 2(x_{14}x_{15} - x_{13}x_{16}) \Omega) x_{13} + (x_9 - ((x_{16})^2 - (x_{13})^2 - (x_{14})^2 + (x_{15})^2) \Omega) x_{16} \end{aligned} \right) \\
& + \lambda_{16} \frac{1}{2} \left(\begin{aligned} & - (x_7 - 2(x_{13}x_{15} + x_{14}x_{16}) \Omega) x_{13} \\ & - (x_8 - 2(x_{14}x_{15} - x_{13}x_{16}) \Omega) x_{14} - (x_9 - ((x_{16})^2 - (x_{13})^2 - (x_{14})^2 + (x_{15})^2) \Omega) x_{15} \end{aligned} \right) \\
& + \lambda_{17} \frac{1}{2} \left(\begin{aligned} & \left(x_{12} - ((x_{20})^2 - (x_{17})^2 - (x_{18})^2 + (x_{19})^2) \Omega \right) x_{18} \\ & - (x_{11} - 2(x_{18}x_{19} - x_{17}x_{20}) \Omega) x_{19} + (x_{10} - 2(x_{17}x_{19} + x_{18}x_{20}) \Omega) x_{20} \end{aligned} \right) \\
& + \lambda_{18} \frac{1}{2} \left(\begin{aligned} & - (x_{12} - ((x_{20})^2 - (x_{17})^2 - (x_{18})^2 + (x_{19})^2) \Omega) x_{17} \\ & + (x_{11} - 2(x_{18}x_{19} - x_{17}x_{20}) \Omega) x_{20} + (x_{10} - 2(x_{17}x_{19} + x_{18}x_{20}) \Omega) x_{19} \end{aligned} \right) \\
& + \lambda_{19} \frac{1}{2} \left(\begin{aligned} & - (x_{10} - 2(x_{17}x_{19} + x_{18}x_{20}) \Omega) x_{18} \\ & + (x_{11} - 2(x_{18}x_{19} - x_{17}x_{20}) \Omega) x_{17} + (x_{12} - ((x_{20})^2 - (x_{17})^2 - (x_{18})^2 + (x_{19})^2) \Omega) x_{20} \end{aligned} \right) \\
& + \lambda_{20} \frac{1}{2} \left(\begin{aligned} & - (x_{10} - 2(x_{17}x_{19} + x_{18}x_{20}) \Omega) x_{17} \\ & - (x_{11} - 2(x_{18}x_{19} - x_{17}x_{20}) \Omega) x_{18} - (x_{12} - ((x_{20})^2 - (x_{17})^2 - (x_{18})^2 + (x_{19})^2) \Omega) x_{19} \end{aligned} \right) \quad (64)
\end{aligned}$$

For the minimum time cost problem defined by Equation (60), the part of the Hamiltonian that depends on the controls, the switching function, turns out to be:

$$H^*(\lambda, \mathbf{x}, \mathbf{u}) = \frac{1}{m} (\lambda_4 f_x + \lambda_5 f_y + \lambda_6 f_z) + \frac{1}{I_{11}^C} \lambda_7 u_4 + \frac{1}{I_{22}^C} \lambda_8 u_5 + \frac{1}{I_{33}^C} \lambda_9 u_6 \quad (65)$$

where

$$\begin{aligned} f_x &= (q_4^{C2} + q_1^{C2} - q_2^{C2} - q_3^{C2}) u_1 + 2(q_1^C q_2^C - q_3^C q_4^C) u_2 + 2(q_1^C q_3^C + q_2^C q_4^C) u_3 \\ f_y &= 2(q_1^C q_2^C + q_3^C q_4^C) u_1 + (q_4^{C2} - q_1^{C2} + q_2^{C2} - q_3^{C2}) u_2 + 2(q_2^C q_3^C - q_1^C q_4^C) u_3 \\ f_z &= 2(q_1^C q_3^C - q_2^C q_4^C) u_1 + 2(q_2^C q_3^C + q_1^C q_4^C) u_2 + (q_4^{C2} - q_1^{C2} - q_2^{C2} + q_3^{C2}) u_3 \end{aligned} \quad (66)$$

This results in Equation (65) written as:

$$H^*(\lambda, \mathbf{x}, \mathbf{u}) = \frac{1}{m} (\text{switch}_1 u_1 + \text{switch}_2 u_2 + \text{switch}_3 u_3) + \frac{1}{I_{11}^C} \lambda_7 u_4 + \frac{1}{I_{22}^C} \lambda_8 u_5 + \frac{1}{I_{33}^C} \lambda_9 u_6 \quad (67)$$

where the switching functions that defined the control above are given by the relations:

$$\begin{aligned} \text{switch}_1 &= \lambda_4 (q_4^{C2} + q_1^{C2} - q_2^{C2} - q_3^{C2}) + \lambda_5 2(q_1^C q_2^C + q_3^C q_4^C) + \lambda_6 2(q_1^C q_3^C - q_2^C q_4^C) \\ \text{switch}_2 &= \lambda_4 2(q_1^C q_2^C - q_3^C q_4^C) + \lambda_5 (q_4^{C2} - q_1^{C2} + q_2^{C2} - q_3^{C2}) + \lambda_6 2(q_2^C q_3^C + q_1^C q_4^C) \\ \text{switch}_3 &= \lambda_4 2(q_1^C q_3^C + q_2^C q_4^C) + \lambda_5 2(q_2^C q_3^C - q_1^C q_4^C) + \lambda_6 (q_4^{C2} - q_1^{C2} - q_2^{C2} + q_3^{C2}) \end{aligned} \quad (68)$$

Assuming there are no singular arcs, since all six controls enter the switching function (Hamiltonian) linearly, the optimal control that minimizes the Hamiltonian in the case of minimum time is the bang-bang control defined by:

$$\begin{aligned} u_1 &= \begin{cases} 1, & \text{switch}_1 < 0 \\ -1, & \text{switch}_1 \geq 0 \end{cases} & u_2 &= \begin{cases} 1, & \text{switch}_2 < 0 \\ -1, & \text{switch}_2 \geq 0 \end{cases} & u_3 &= \begin{cases} 1, & \text{switch}_3 < 0 \\ -1, & \text{switch}_3 \geq 0 \end{cases} \\ u_4 &= \begin{cases} 1, & \lambda_7 < 0 \\ -1, & \lambda_7 \geq 0 \end{cases} & u_5 &= \begin{cases} 1, & \lambda_8 < 0 \\ -1, & \lambda_8 \geq 0 \end{cases} & u_6 &= \begin{cases} 1, & \lambda_9 < 0 \\ -1, & \lambda_9 \geq 0 \end{cases} \end{aligned} \quad (69)$$

Likewise, developing the Hamiltonian for the minimum quadratic-control PI based on the running cost (61), the switching function becomes:

$$\begin{aligned}
H^*(\lambda, \mathbf{x}, \mathbf{u}) = & \frac{1}{2} (u_1^2 + u_2^2 + u_3^2 + u_4^2 + u_5^2 + u_6^2) \\
& + \frac{1}{m} (\lambda_4 f_1 + \lambda_5 f_2 + \lambda_6 f_3) + \frac{1}{I_{11}^C} \lambda_7 u_4 + \frac{1}{I_{22}^C} \lambda_8 u_5 + \frac{1}{I_{33}^C} \lambda_9 u_6.
\end{aligned} \tag{70}$$

Taking into account Equation (66), the resulting optimal control that minimizes the Hamiltonian for the minimum quadratic-control (minimum energy) PI becomes:

$$\begin{aligned}
u_i = & \begin{cases} 1, & \frac{\text{switch}_i}{m} < -1 \\ -\frac{\text{switch}_i}{m}, & -1 \leq \frac{\text{switch}_i}{m} \leq 1 \quad i = 1, 2, 3 \\ -1, & \frac{\text{switch}_i}{m} > 1 \end{cases} \\
\text{and } u_i = & \begin{cases} 1, & \frac{\lambda_{i+3}}{I_{jj}^C} < -1 \\ -\frac{\lambda_{i+3}}{I_{jj}^C}, & -1 \leq \frac{\lambda_{i+3}}{I_{jj}^C} \leq 1 \quad i = 4, 5, 6, \quad j = i - 3 \\ -1, & \frac{\lambda_{i+3}}{I_{jj}^C} > 1 \end{cases}.
\end{aligned} \tag{71}$$

For the minimum fuel case, corresponding to the running cost in Equation (62), the switching function can be written as:

$$\begin{aligned}
H^*(\lambda, \mathbf{x}, \mathbf{u}) = & (|u_1| + |u_2| + |u_3| + |u_4| + |u_5| + |u_6|) \\
& + \frac{1}{m} (\lambda_4 f_1 + \lambda_5 f_2 + \lambda_6 f_3) + \frac{1}{I_{11}^C} \lambda_7 u_4 + \frac{1}{I_{22}^C} \lambda_8 u_5 + \frac{1}{I_{33}^C} \lambda_9 u_6
\end{aligned} \tag{72}$$

and, therefore, the optimal control structure becomes:

$$\begin{aligned}
u_i &= \begin{cases} 1, & \text{switch}_i < -m \\ 0, & -m \leq \text{switch}_i \leq m \quad i = 1, 2, 3 \\ -1, & \text{switch}_i > m \end{cases} \\
\text{and } u_i &= \begin{cases} 1, & \lambda_{i+3} < -I_{jj}^C \\ 0, & -I_{jj}^C \leq \lambda_{i+3} \leq I_{jj}^C \quad i = 4, 5, 6, \quad j = i - 3 \\ -1, & \lambda_{i+3} > I_{jj}^C \end{cases} .
\end{aligned} \tag{73}$$

The differential equations for costates are obtained via differentiating the Hamiltonian:

$$\left(\frac{\delta H}{\delta \mathbf{x}} \right)^T = -\dot{\lambda} \tag{74}$$

which yields:

$$\begin{aligned}
\dot{\lambda}_1 &= -3\lambda_4\Omega^2m^{-1}, \quad \dot{\lambda}_2 = 0, \quad \dot{\lambda}_3 = \lambda_6\Omega^2m^{-1} \\
\dot{\lambda}_4 &= -\lambda_1 + 2\lambda_5\Omega m^{-1}, \quad \dot{\lambda}_5 = -\lambda_2 - 2\lambda_4\Omega m^{-1} . \\
\dot{\lambda}_6 &= -\lambda_3
\end{aligned} \tag{75}$$

The next six adjoint equations, corresponding to costates 7–12, take the form of Equations (76)–(81):

$$\dot{\lambda}_7 = \frac{I_{11}^C - I_{33}^C}{I_{22}^C} \lambda_8 x_9 + \frac{I_{22}^C - I_{11}^C}{I_{33}^C} \lambda_9 x_8 - \lambda_{13} \frac{1}{2} \Omega x_{16} - \lambda_{14} \frac{1}{2} \Omega x_{15} + \lambda_{15} \frac{1}{2} \Omega x_{14} + \lambda_{16} \frac{1}{2} \Omega x_{13} \tag{76}$$

$$\dot{\lambda}_8 = \frac{I_{33}^C - I_{22}^C}{I_{11}^C} \lambda_7 x_9 + \frac{I_{22}^C - I_{11}^C}{I_{33}^C} \lambda_9 x_7 + \lambda_{13} \frac{1}{2} \Omega x_{15} - \lambda_{14} \frac{1}{2} \Omega x_{16} - \lambda_{15} \frac{1}{2} \Omega x_{13} + \lambda_{16} \frac{1}{2} \Omega x_{14} \tag{77}$$

$$\dot{\lambda}_9 = \frac{I_{33}^C - I_{22}^C}{I_{11}^C} \lambda_7 x_8 + \frac{I_{11}^C - I_{33}^C}{I_{22}^C} \lambda_8 x_7 - \lambda_{13} \frac{1}{2} \Omega x_{14} + \lambda_{14} \frac{1}{2} \Omega x_{13} - \lambda_{15} \frac{1}{2} \Omega x_{16} + \lambda_{16} \frac{1}{2} \Omega x_{15} \tag{78}$$

$$\dot{\lambda}_{10} = \frac{I_{11}^T - I_{33}^T}{I_{22}^T} \lambda_{11} x_{12} + \frac{I_{22}^T - I_{11}^T}{I_{33}^T} \lambda_{12} x_{11} - \lambda_{17} \frac{1}{2} \Omega x_{20} - \lambda_{18} \frac{1}{2} \Omega x_{19} + \lambda_{19} \frac{1}{2} \Omega x_{18} + \lambda_{20} \frac{1}{2} \Omega x_{17} \tag{79}$$

$$\dot{\lambda}_{11} = \frac{I_{33}^T - I_{22}^T}{I_{11}^T} \lambda_{10} x_{12} + \frac{I_{22}^T - I_{11}^T}{I_{33}^T} \lambda_{12} x_{10} + \lambda_{17} \frac{1}{2} \Omega x_{19} - \lambda_{18} \frac{1}{2} \Omega x_{20} - \lambda_{19} \frac{1}{2} \Omega x_{17} + \lambda_{20} \frac{1}{2} \Omega x_{18} \tag{80}$$

$$\dot{\lambda}_{12} = \frac{I_{33}^T - I_{22}^T}{I_{11}^T} \lambda_{10} x_{11} + \frac{I_{11}^T - I_{33}^T}{I_{22}^T} \lambda_{11} x_{10} - \lambda_{17} \frac{1}{2} \Omega x_{18} + \lambda_{18} \frac{1}{2} \Omega x_{17} - \lambda_{19} \frac{1}{2} \Omega x_{20} + \lambda_{20} \frac{1}{2} \Omega x_{19} \quad (81)$$

The next four equations, corresponding to costates 13–16, are of the form of Equations (82)–(85):

$$\begin{aligned} \dot{\lambda}_{13} = & -\frac{1}{m} \left(\lambda_4 (2x_{13}u_1 + 2x_{14}u_2 + 2x_{15}u_3) + \lambda_5 (2x_{14}u_1 - 2x_{13}u_2 - 2x_{16}u_3) \right) \\ & + \lambda_6 (2x_{15}u_1 + 2x_{16}u_2 - 2x_{13}u_3) \\ & - \lambda_{13} \frac{1}{2} (2x_{13}\Omega x_{14} - 2x_{16}\Omega x_{15} - 2x_{15}\Omega x_{16}) \\ & - \lambda_{14} \frac{1}{2} \left(-\left(x_9 - (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2) \Omega \right) - 2x_{13}^2 \Omega - 2x_{16}^2 \Omega - 2x_{15}^2 \Omega \right) \\ & - \lambda_{15} \frac{1}{2} \left(2x_{15}\Omega x_{14} + \left(x_8 - 2(x_{14}x_{15} - x_{13}x_{16}) \Omega \right) + 2x_{16}\Omega x_{13} + 2x_{13}\Omega x_{16} \right) \\ & - \lambda_{16} \frac{1}{2} \left(2x_{15}\Omega x_{13} - \left(x_7 - 2(x_{13}x_{15} + x_{14}x_{16}) \Omega \right) - 2x_{16}\Omega x_{14} - 2x_{13}\Omega x_{15} \right) \end{aligned} \quad (82)$$

$$\begin{aligned} \dot{\lambda}_{14} = & -\frac{1}{m} \left(\lambda_4 (-2x_{14}u_1 + 2x_{13}u_2 + 2x_{16}u_3) + \lambda_5 (2x_{13}u_1 + 2x_{14}u_2 + 2x_{15}u_3) \right) \\ & + \lambda_6 (-2x_{16}u_1 + 2x_{15}u_2 - 2x_{14}u_3) \\ & - \lambda_{13} \frac{1}{2} \left(2x_{14}^2 \Omega + \left(x_9 - (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2) \Omega \right) + 2x_{16}^2 \Omega + 2x_{15}^2 \Omega \right) \\ & - \lambda_{14} \frac{1}{2} (2x_{13}\Omega x_{14} - 2x_{16}\Omega x_{15} - 2x_{15}\Omega x_{16}) \\ & - \lambda_{15} \frac{1}{2} \left(-\left(x_7 - 2(x_{13}x_{15} + x_{14}x_{16}) \Omega \right) + 2x_{16}\Omega x_{14} - 2x_{15}\Omega x_{13} + 2x_{14}\Omega x_{16} \right) \\ & - \lambda_{16} \frac{1}{2} \left(2x_{16}\Omega x_{13} - \left(x_8 - 2(x_{14}x_{15} - x_{13}x_{16}) \Omega \right) + 2x_{15}\Omega x_{14} - 2x_{14}\Omega x_{15} \right) \end{aligned} \quad (83)$$

$$\begin{aligned}
\dot{\lambda}_{15} = & -\frac{1}{m} \left(\lambda_4 (-2x_{15}u_1 - 2x_{16}u_2 + 2x_{13}u_3) + \lambda_5 (2x_{16}u_1 - 2x_{15}u_2 + 2x_{14}u_3) \right) \\
& + \lambda_6 (2x_{13}u_1 + 2x_{14}u_2 + 2x_{15}u_3) \\
& - \lambda_{13} \frac{1}{2} \left(-2x_{15}\Omega x_{14} - (x_8 - 2(x_{14}x_{15} - x_{13}x_{16})\Omega) + 2x_{15}\Omega x_{14} - 2x_{13}\Omega x_{16} \right) \\
& - \lambda_{14} \frac{1}{2} \left(2x_{15}\Omega x_{13} - 2x_{16}\Omega x_{14} + (x_7 - 2(x_{13}x_{15} + x_{14}x_{16})\Omega) - 2x_{13}\Omega x_{15} \right) \\
& - \lambda_{15} \frac{1}{2} \left(2x_{13}\Omega x_{14} - 2x_{14}\Omega x_{13} + 2x_{15}\Omega x_{16} \right) \\
& - \lambda_{16} \frac{1}{2} \left(+2x_{13}^2\Omega + 2x_{14}^2\Omega - (x_9 - (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2)\Omega) + 2x_{15}^2\Omega \right)
\end{aligned} \tag{84}$$

$$\begin{aligned}
\dot{\lambda}_{16} = & -\frac{1}{m} \left(\lambda_4 (2x_{16}u_1 - 2x_{15}u_2 + 2x_{14}u_3) + \lambda_5 (2x_{15}u_1 + 2x_{16}u_2 - 2x_{13}u_3) \right) \\
& + \lambda_6 (-2x_{14}u_1 + 2x_{13}u_2 + 2x_{16}u_3) \\
& - \lambda_{13} \frac{1}{2} \left(-2x_{16}\Omega x_{14} - 2x_{13}\Omega x_{15} + (x_7 - 2(x_{13}x_{15} + x_{14}x_{16})\Omega) - 2x_{16}\Omega x_{14} \right) \\
& - \lambda_{14} \frac{1}{2} \left(2x_{16}\Omega x_{13} + 2x_{16}\Omega x_{13} + (x_8 - 2(x_{14}x_{15} - x_{13}x_{16})\Omega) - 2x_{14}\Omega x_{15} \right) \\
& - \lambda_{15} \frac{1}{2} \left(2x_{13}^2\Omega + 2x_{14}^2\Omega - (x_9 - (x_{16}^2 - x_{13}^2 - x_{14}^2 + x_{15}^2)\Omega) + 2x_{16}^2\Omega \right) \\
& - \lambda_{16} \frac{1}{2} \left(2x_{13}\Omega x_{14} - 2x_{13}\Omega x_{14} + 2x_{15}\Omega x_{16} \right)
\end{aligned} \tag{85}$$

and, the final four adjoint equations for costates 17–20, take the form of Equations (86)–(89):

$$\begin{aligned}
\dot{\lambda}_{17} = & -\lambda_{17} \frac{1}{2} \left(2x_{17}\Omega x_{18} - 2x_{20}\Omega x_{19} - 2x_{19}\Omega x_{20} \right) \\
& - \lambda_{18} \frac{1}{2} \left(- (x_{12} - (x_{20}^2 - x_{17}^2 - x_{18}^2 + x_{19}^2)\Omega) - 2x_{17}^2\Omega - 2x_{20}^2\Omega - 2x_{19}^2\Omega \right) \\
& - \lambda_{19} \frac{1}{2} \left(2x_{19}\Omega x_{18} + (x_{11} - 2(x_{18}x_{19} - x_{17}x_{20})\Omega) + 2x_{20}\Omega x_{17} + 2x_{17}\Omega x_{20} \right) \\
& - \lambda_{20} \frac{1}{2} \left(2x_{19}\Omega x_{17} - (x_{10} - 2(x_{17}x_{19} + x_{18}x_{20})\Omega) - 2x_{20}\Omega x_{18} - 2x_{17}\Omega x_{19} \right)
\end{aligned} \tag{86}$$

$$\begin{aligned}
\dot{\lambda}_{18} = & -\lambda_{17} \frac{1}{2} \left(2x_{18}^2 \Omega + \left(x_{12} - (x_{20}^2 - x_{17}^2 - x_{18}^2 + x_{19}^2) \Omega \right) + 2x_{20}^2 \Omega + 2x_{19}^2 \Omega \right) \\
& - \lambda_{18} \frac{1}{2} \left(2x_{17} \Omega x_{18} - 2x_{20} \Omega x_{19} - 2x_{19} \Omega x_{20} \right) \\
& - \lambda_{19} \frac{1}{2} \left(- \left(x_{10} - 2(x_{17}x_{19} + x_{18}x_{20}) \Omega \right) + 2x_{20} \Omega x_{18} - 2x_{19} \Omega x_{17} + 2x_{18} \Omega x_{20} \right) \\
& - \lambda_{20} \frac{1}{2} \left(2x_{20} \Omega x_{17} - \left(x_{11} - 2(x_{14}x_{19} - x_{17}x_{20}) \Omega \right) + 2x_{19} \Omega x_{18} - 2x_{18} \Omega x_{19} \right)
\end{aligned} \tag{87}$$

$$\begin{aligned}
\dot{\lambda}_{19} = & -\lambda_{17} \frac{1}{2} \left(-2x_{19} \Omega x_{18} - \left(x_{11} - 2(x_{18}x_{19} - x_{17}x_{20}) \Omega \right) + 2x_{19} \Omega x_{18} - 2x_{17} \Omega x_{20} \right) \\
& - \lambda_{18} \frac{1}{2} \left(2x_{19} \Omega x_{17} - 2x_{20} \Omega x_{18} + \left(x_{10} - 2(x_{17}x_{19} + x_{18}x_{20}) \Omega \right) - 2x_{17} \Omega x_{19} \right) \\
& - \lambda_{19} \frac{1}{2} \left(2x_{17} \Omega x_{18} - 2x_{18} \Omega x_{17} + 2x_{19} \Omega x_{20} \right) \\
& - \lambda_{20} \frac{1}{2} \left(+2x_{17}^2 \Omega + 2x_{18}^2 \Omega - \left(x_{12} - (x_{20}^2 - x_{17}^2 - x_{18}^2 + x_{19}^2) \Omega \right) + 2x_{19}^2 \Omega \right)
\end{aligned} \tag{88}$$

$$\begin{aligned}
\dot{\lambda}_{20} = & -\lambda_{17} \frac{1}{2} \left(-2x_{20} \Omega x_{18} - 2x_{17} \Omega x_{19} + \left(x_{10} - 2(x_{17}x_{19} + x_{18}x_{20}) \Omega \right) - 2x_{20} \Omega x_{18} \right) \\
& - \lambda_{18} \frac{1}{2} \left(2x_{20} \Omega x_{17} + 2x_{20} \Omega x_{17} + \left(x_{11} - 2(x_{18}x_{19} - x_{17}x_{20}) \Omega \right) - 2x_{18} \Omega x_{19} \right) \\
& - \lambda_{19} \frac{1}{2} \left(2x_{17}^2 \Omega + 2x_{18}^2 \Omega - \left(x_{12} - (x_{20}^2 - x_{17}^2 - x_{18}^2 + x_{19}^2) \Omega \right) + 2x_{20}^2 \Omega \right) \\
& - \lambda_{20} \frac{1}{2} \left(2x_{17} \Omega x_{18} - 2x_{17} \Omega x_{18} + 2x_{19} \Omega x_{20} \right)
\end{aligned} \tag{89}$$

The form shown in Equations (75)–(89) is different than presented in the previous chapter, again because of the body-mounted thrusters.

The next step is to address the transversality conditions, which with account to the following coupled terminal variations, followed directly from Equations (55)–(58). The ten scalar equations, e_1, e_2, \dots, e_{10} , represent the matching of translational and angular position and velocity of the docking points in the three components of the orbital coordinate frame at the time t_f based on Equations (57) and (58). Variable E represents an endpoint cost that is set to zero for both the minimum time and minimum-control

scenarios (Yan et al. 2002) and are equivalent to Equations (26) and (27) of Chapter II. The first six equations result in expressions that only contain the parameters ν_i , $i=1,\dots,6$.

$$\lambda_1(t_f) = -\nu_1, \lambda_2(t_f) = -\nu_2, \lambda_3(t_f) = -\nu_3, \lambda_4(t_f) = -\nu_4, \lambda_5(t_f) = -\nu_5, \lambda_6(t_f) = -\nu_6 \quad (90)$$

Realizing the specific relationships, such as:

$$\left. \frac{\delta \bar{E}}{\delta x_7} \right|_{t_f} = \lambda_{7,t_f} = f_7(\mathbf{x}_{t_f}, \boldsymbol{\lambda}_{t_f}) + \nu_7 \text{ and } \left. \frac{\delta \bar{E}}{\delta x_{10}} \right|_{t_f} = \lambda_{10,t_f} = f_{10}(\mathbf{x}_{t_f}, \boldsymbol{\lambda}_{t_f}) - \nu_7 \quad (91)$$

and combining terms, seven more conditions are developed that must be satisfied, along with six docking states for position and velocity given by Equations (57) and (58) and the seven angular states based on Equations (55) and (56), at t_f . These conditions are shown in Equation (92):

$$\begin{aligned} \zeta_1(t_f) &= \lambda_{7,t_f} + \lambda_{10,t_f} + f_7(\mathbf{x}_{t_f}, \boldsymbol{\lambda}_{t_f}) + f_{10}(\mathbf{x}_{t_f}, \boldsymbol{\lambda}_{t_f}) = e_{14}(t_f) = 0 \\ &\vdots \\ \zeta_6(t_f) &= \lambda_{i,t_f} + \lambda_{(i+3)t_f} + f_i(\mathbf{x}_{t_f}, \boldsymbol{\lambda}_{t_f}) + f_{(i+3)}(\mathbf{x}_{t_f}, \boldsymbol{\lambda}_{t_f}) = e_{(i+7)}(t_f) = 0, \text{ for } i = 7, 8, 9 \text{ and} \\ \zeta_7(t_f) &= \lambda_{i,t_f} + \lambda_{(i+4)t_f} + f_i(\mathbf{x}_{t_f}, \boldsymbol{\lambda}_{t_f}) + f_{(i+4)}(\mathbf{x}_{t_f}, \boldsymbol{\lambda}_{t_f}) = e_{(i+4)}(t_f) = 0, \text{ for } i = 13, \dots, 16 \end{aligned} \quad (92)$$

where $f_i(\mathbf{x}_{t_f}, \boldsymbol{\lambda}_{t_f})$ is the portion of $\left(\frac{\partial \bar{E}}{\partial \mathbf{x}} \right)^T \bigg|_{t=t_f}$ that only depends on \mathbf{x}_{t_f} and $\boldsymbol{\lambda}_{t_f}$. This

results in the remaining final conditions, ζ , along with Equations (55)–(58) that need to be satisfied for an optimal trajectory, $\zeta = [\zeta_1(t_f), \dots, \zeta_7(t_f)]^T = \mathbf{0}$.

Other parameters for the scenario are $m = 1 \text{ kg}$, $\Omega = 0.005 \text{ rad/s}$, $\mathbf{I}^T = \text{diag}([3, 1, 2])$, $\mathbf{I}^C = \mathbf{1}_{3 \times 3}$ (where $\mathbf{1}_{3 \times 3}$ is an identity matrix) with inertia units of $\text{kg} \cdot \text{m}^2$ and t_f set to a maximum of 10 seconds. This scenario initializes the chaser CM starting at a distance of 5 m from the target CM with the target having an initial angular velocity of 0.25 rad/s in both y and z body axis. The body coordinate frames of each spacecraft and the orbit frame are assumed to be coincident with the inertial frame at the beginning of the

simulation. The docking point of the chaser spacecraft is located at 0.25 m in the negative x body frame and the docking position of the target is located at 1m in the positive x body frame. Therefore, the orientations of the two spacecraft will be coincident at docking. The resulting values of the initial states for computer simulations discussed in the following sections are taken from Table 1 in the previous chapter.

C. SOLVING THE PROBLEM NUMERICALLY USING A PSEUDOSPECTRAL METHOD

The optimal control problem posted in Chapter III.A was again solved using GPOPS (Huntington and Rao 2008). This software solves for the optimal control, state and costate history based on a given PI and constraints. As opposed to the 150 node solutions of the last chapter, 200 nodes were chosen for the solution to be consistent with previous research on this topic (Ma et al. 2007).

1. Minimum Time

The 3D trajectory and 2D projections of the trajectories are shown in Figures 33 and 34, respectively. In these figures, the solid line with the square markers shows the trajectory of the docking point aboard the target RSO, the dotted-dashed line with a circular marker represents the chaser spacecraft's CM, and the line with upside-down triangles depicts the trajectory of the docking point aboard the chaser spacecraft. The overlaid sphere is centered at the target RSO's CM and has a radius equal to that of the distance the docking point is offset from its CM.

The resulting control history is shown in Figure 35. Figures 35–38 show the state and costate histories. Figures 39 and 40 show the final difference in endpoint conditions approaching zero and Figure 41 shows the time history of the transversality conditions, Equation (92), as they approach zero. The Hamiltonian is shown in Figure 42. The initial values of the calculated costates are shown in Table 11. Some more details about this solution and required computer resources will also be provided in Chapter III.C.4 when comparing this solution with others.

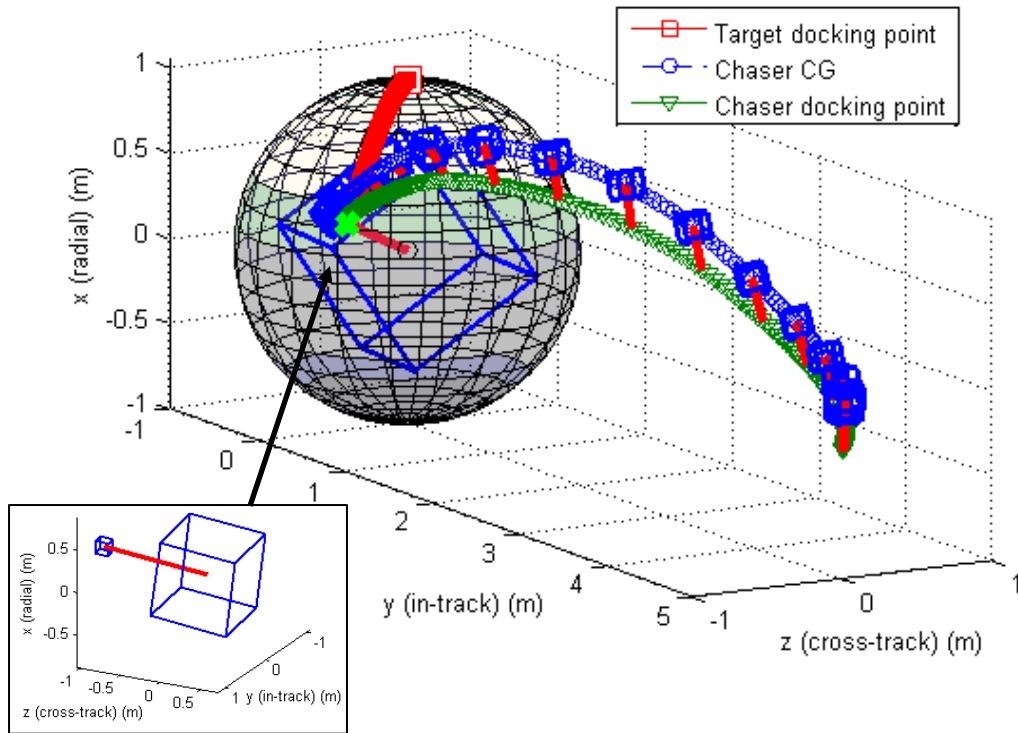


Figure 33. Minimum time solution (GPOPS): 3D plot of optimal rendezvous trajectory.

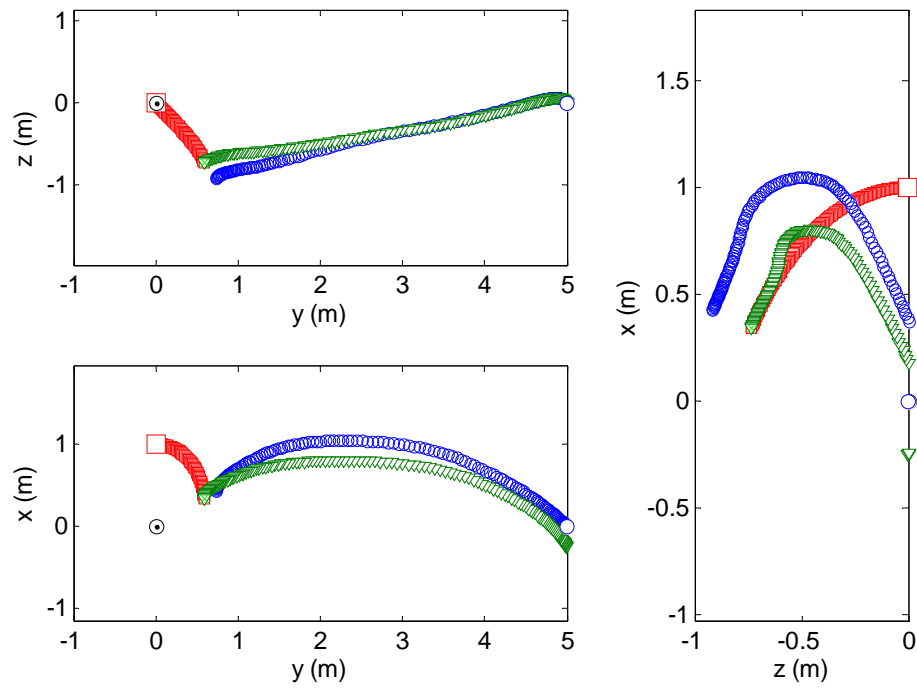


Figure 34. Minimum time solution (GPOPS): 2D plots of optimal rendezvous trajectory.

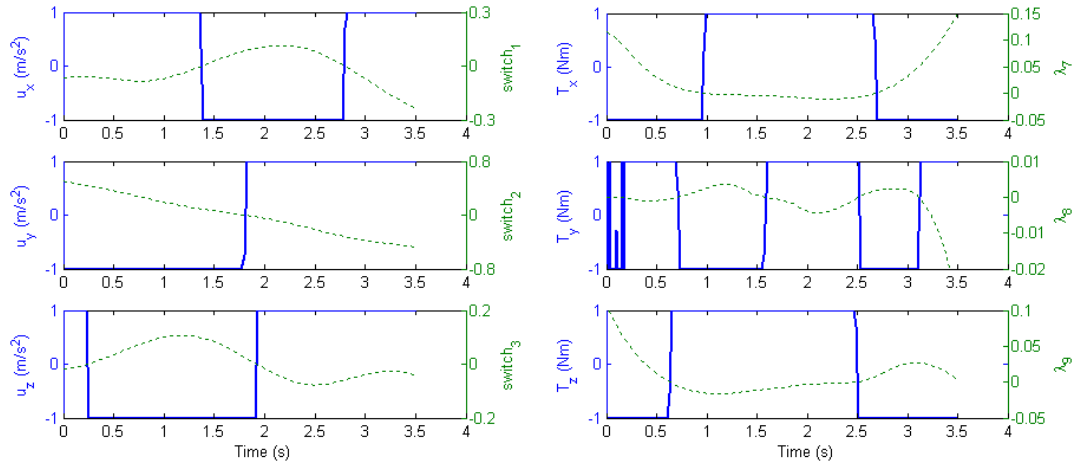


Figure 35. Minimum time solution (GPOPS): control history solution.

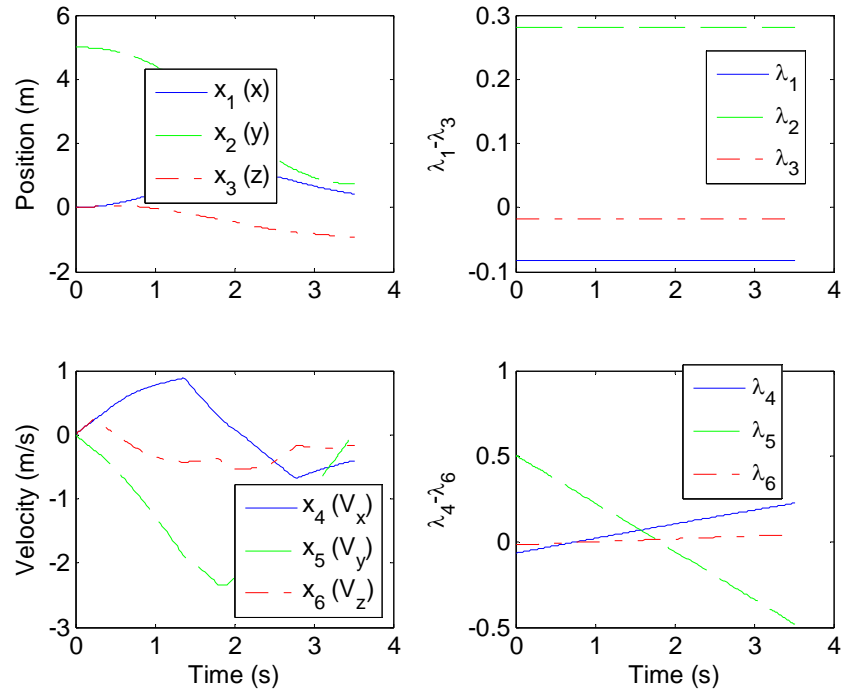


Figure 36. Minimum time solution (GPOPS): state and costate histories for the defining translational variables of the chaser vehicle.

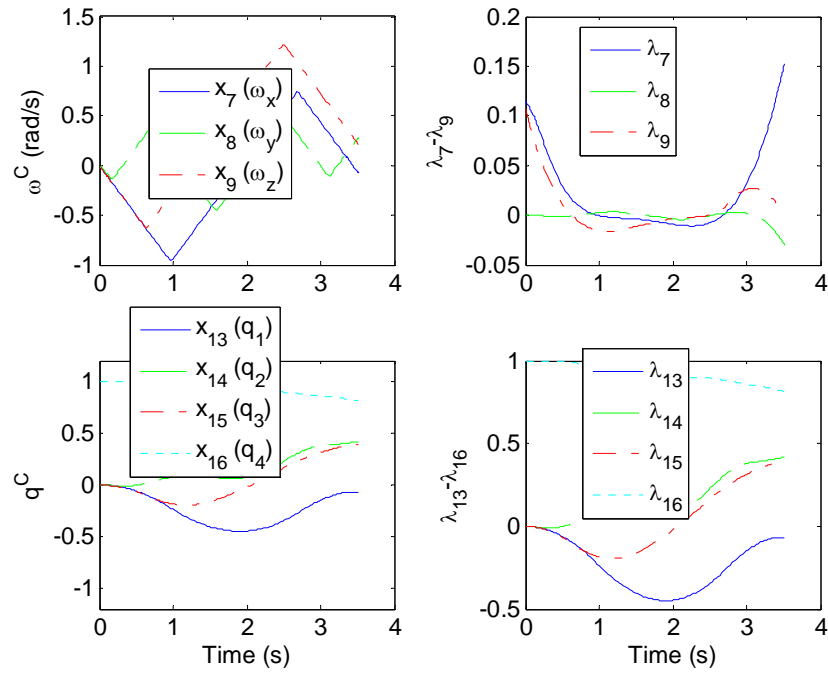


Figure 37. Minimum time solution (GPOPS): state and costate histories for the defining angular parameters of the chaser vehicle.

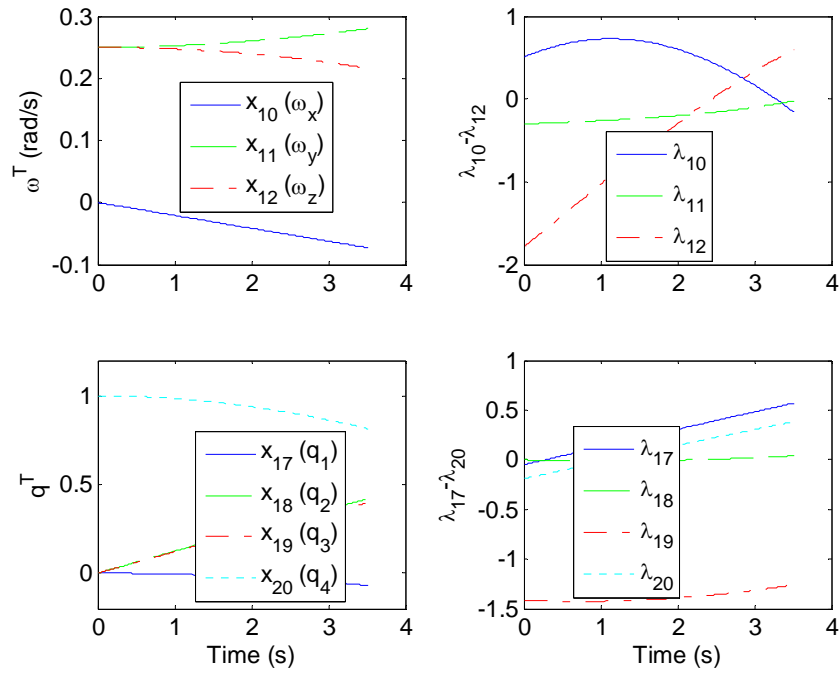


Figure 38. Minimum time solution (GPOPS): state and costate histories for the defining angular parameters of the RSO.

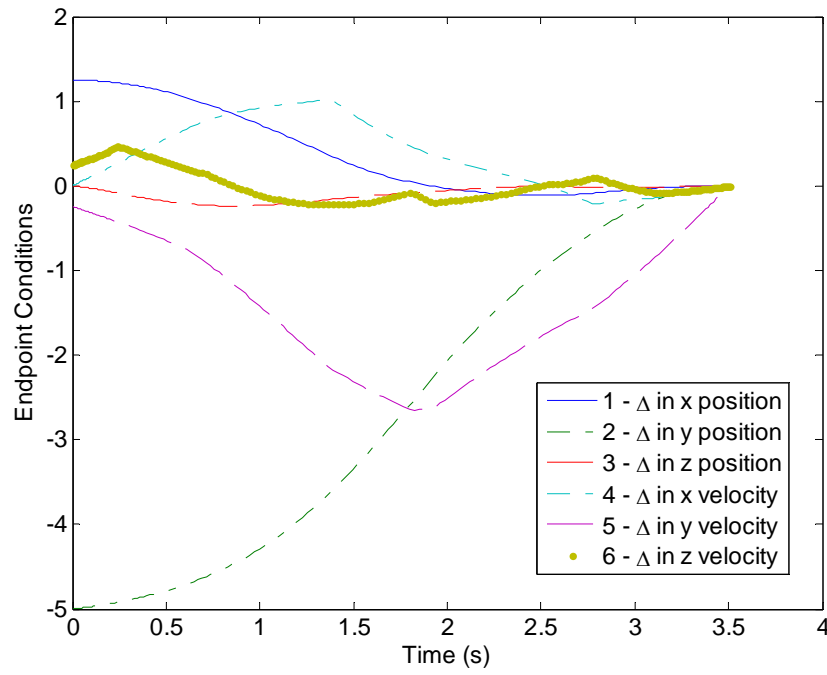


Figure 39. Minimum time solution (GPOPS): time history of the translational endpoint discrepancies.

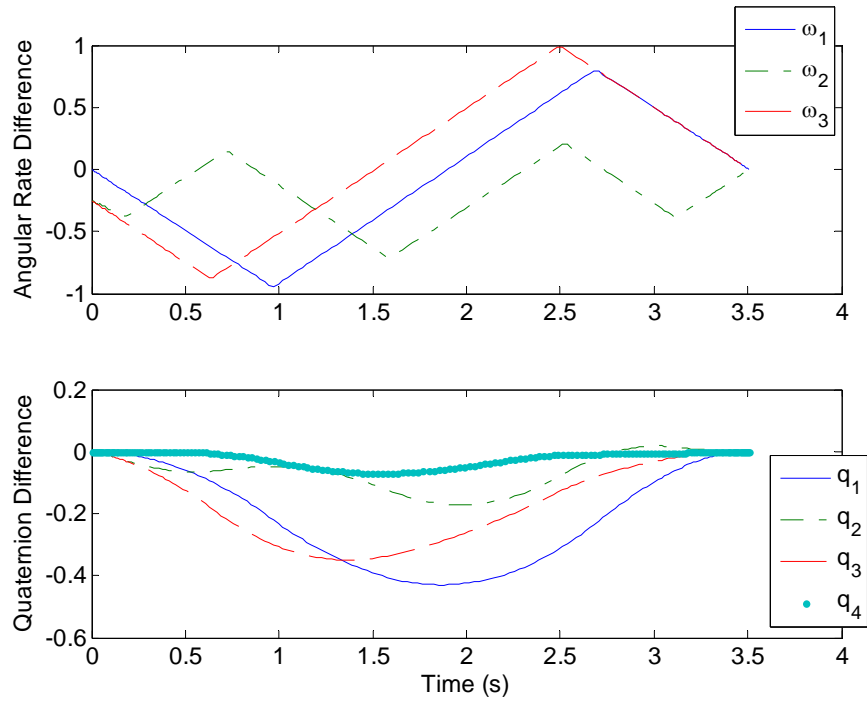


Figure 40. Minimum time solution (GPOPS): time history of the attitude endpoint discrepancies.

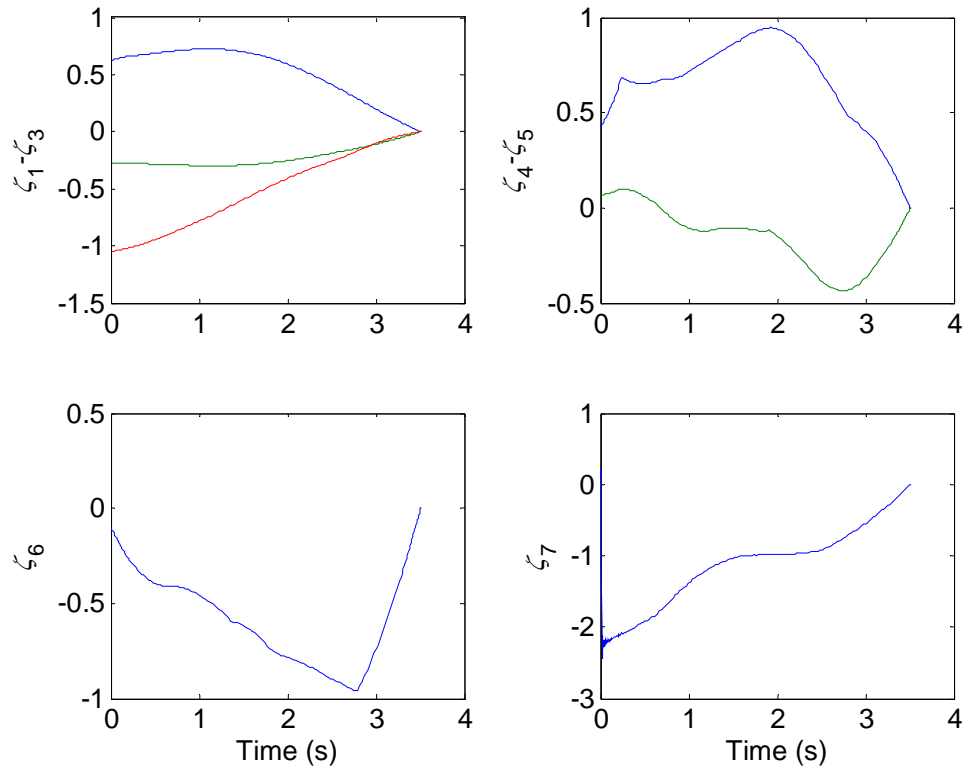


Figure 41. Minimum time solution (GPOPS): history of transversality conditions.

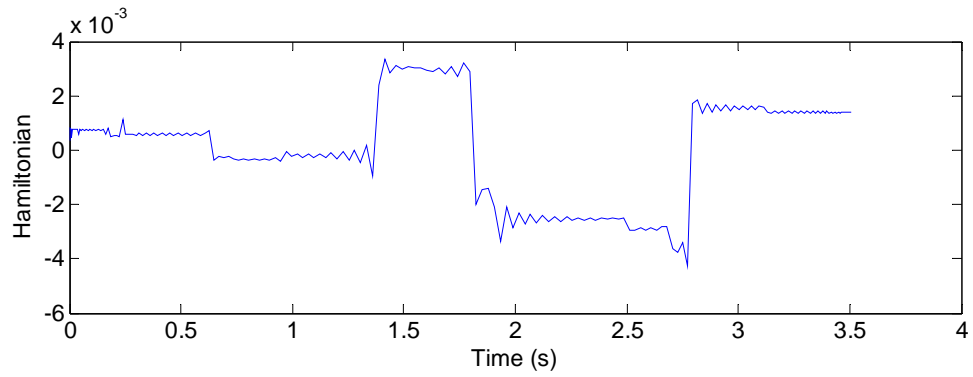


Figure 42. Minimum time solution (GPOPS): history of the Hamiltonian.

<i>Costate</i>	<i>Initial Condition</i>	<i>Costate</i>	<i>Initial Condition</i>	<i>Costate</i>	<i>Initial Condition</i>
λ_1	-0.0832	λ_8	-0.0300	λ_{15}	1.4758
λ_2	0.2801	λ_9	0.0027	λ_{16}	1.9302
λ_3	-0.0172	λ_{10}	-0.1521	λ_{17}	0.7304
λ_4	0.2251	λ_{11}	-0.0245	λ_{18}	-0.9484
λ_5	-0.4795	λ_{12}	0.5898	λ_{19}	-2.1946
λ_6	0.0426	λ_{13}	-0.7500	λ_{20}	-1.5493
λ_7	0.1521	λ_{14}	1.0625	t_f	3.4237

Table 11. The initial values of costates and t_f as defined by GPOPS for minimum time solution.

The important thing about this solution is that there is no singular control in variable u_z as there was in the previous scenario described in Chapter II.E.1, where the translational controls were expressed in the orbital frame rather than in the body frame. However, the solution for T_y might suggest that there may be a singular arc in the beginning of the maneuver. It should be noted that in case of a singular control, the GPOPS control output may not be feasible for onboard implementation due to its highly oscillating nature (Boyarko, Yakimenko and Romano 2009a).

2. Minimum Quadratic Control (Minimum Energy)

For the minimum energy solution, Figures 43 and 44 represent the resulting trajectory in both three dimensions and planar view. Figure 45 shows a plot of the resulting control history, as well as the associated switching function that is used for calculation of the control based on the running cost in Equation (61). Figures 46–48 show the states and costates. The discrepancies in endpoint conditions are shown to reach zero at t_f in Figures 49 and 50. The transversality conditions are shown in Figure 51 and the Hamiltonian is shown in Figure 52. Table 12 summarizes the initial values of the costates that were solved for by the GPOPS solution. The PI of the solution was $J = 0.2445$.

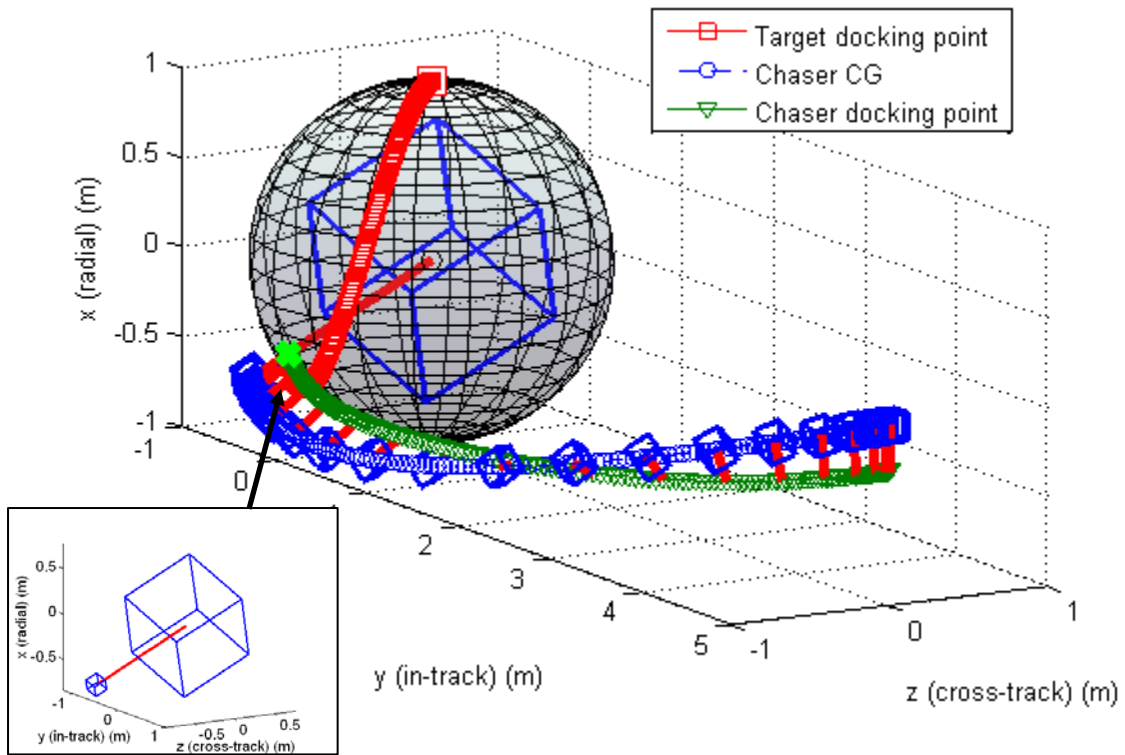


Figure 43. Minimum energy solution (GPOPS): 3D plot of optimal trajectory.

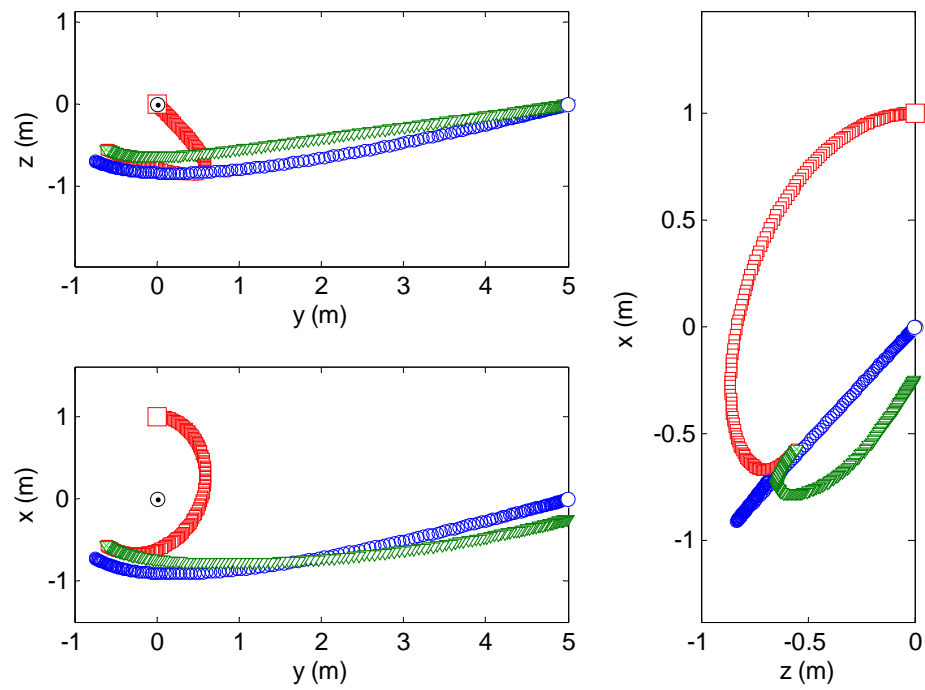


Figure 44. Minimum energy solution (GPOPS): 2D plots of optimal trajectory.

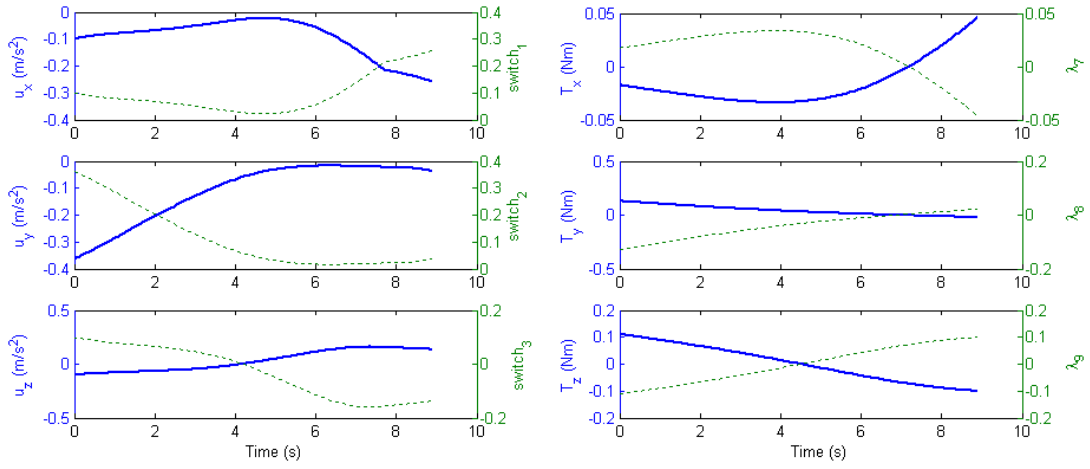


Figure 45. Minimum energy solution (GPOPS): control history with respect to the optimal trajectory.

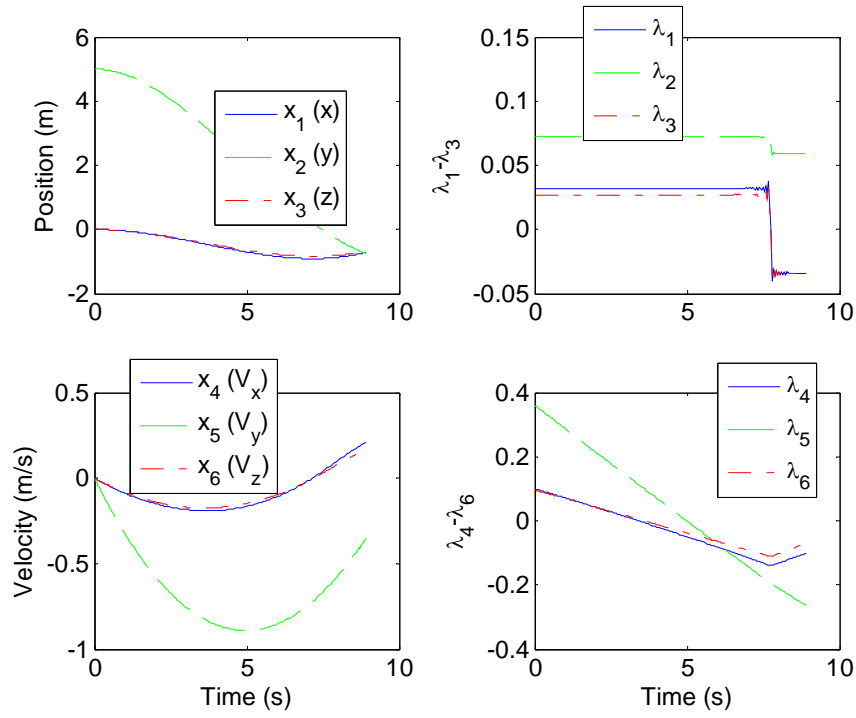


Figure 46. Minimum energy solution (GPOPS): state and costate histories for the translational variables of the chaser vehicle.

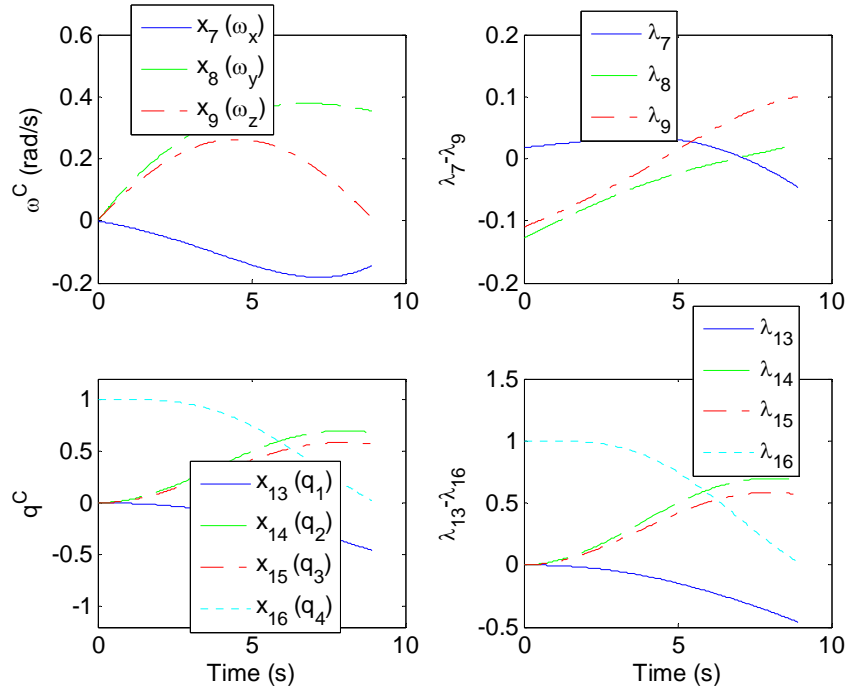


Figure 47. Minimum energy solution (GPOPS): state and costate histories for the defining angular parameters of the chaser vehicle.

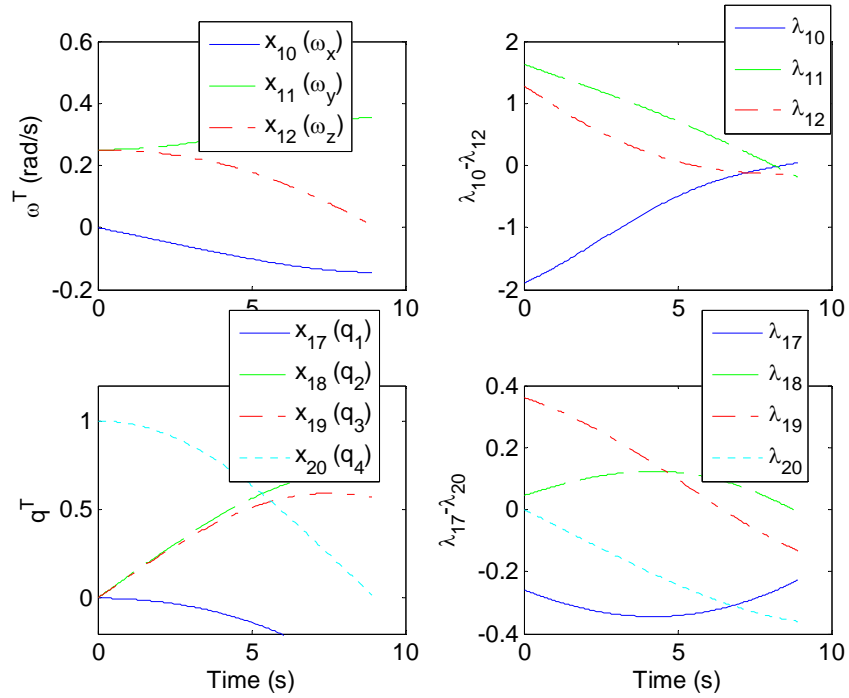


Figure 48. Minimum energy solution (GPOPS): state and costate histories for the defining angular parameters of the RSO.

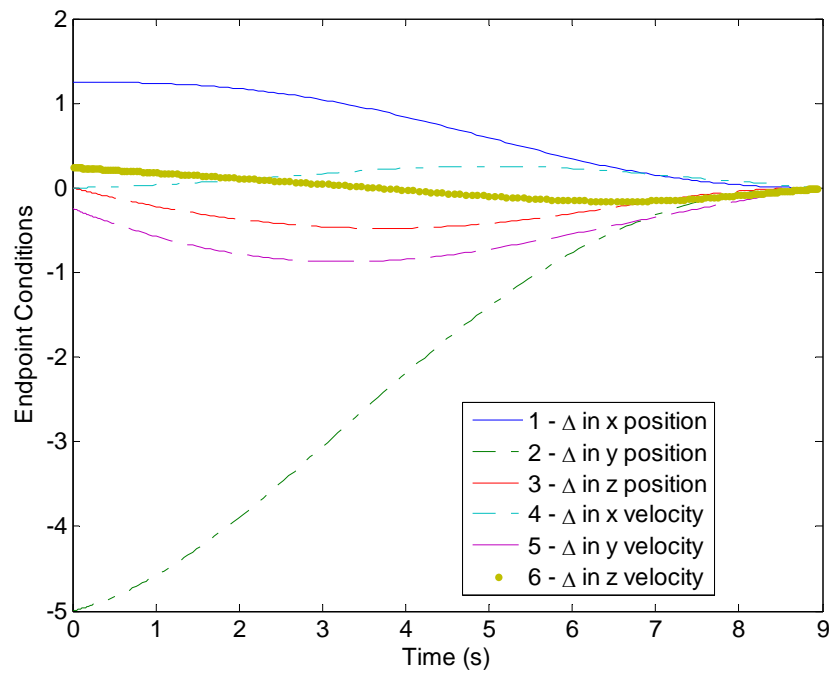


Figure 49. Minimum energy solution (GPOPS): history of the translational endpoint discrepancies.

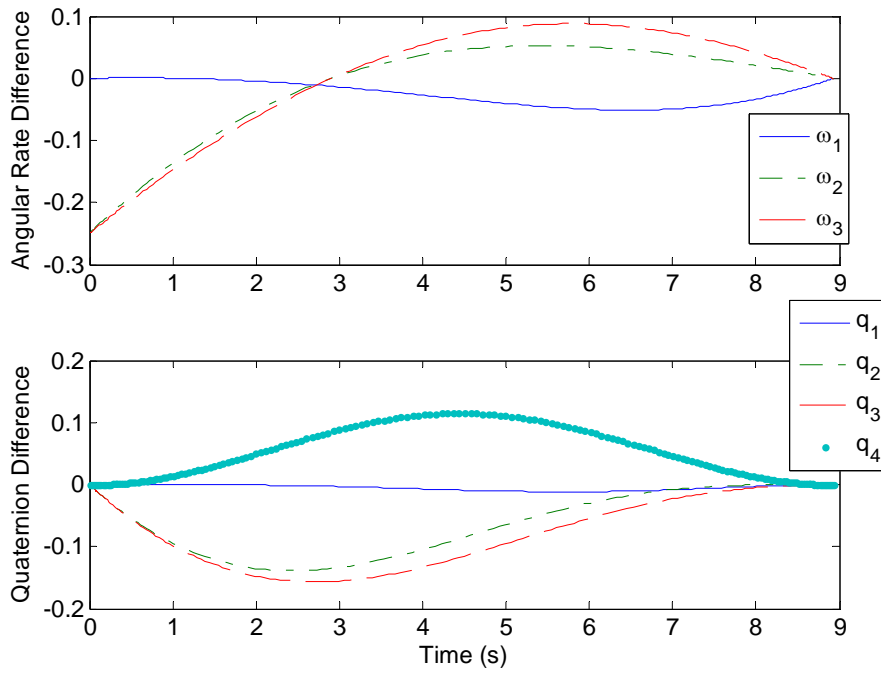


Figure 50. Minimum energy solution (GPOPS): history of the rotational endpoint discrepancies.

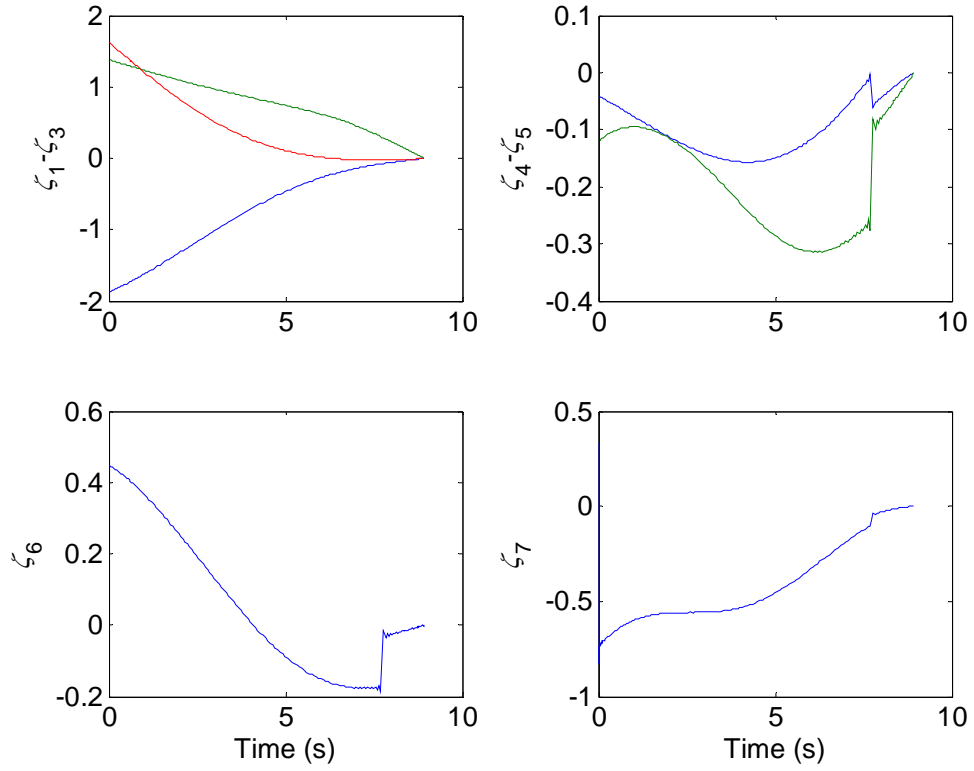


Figure 51. Minimum energy solution (GPOPS): history of the transversality conditions.

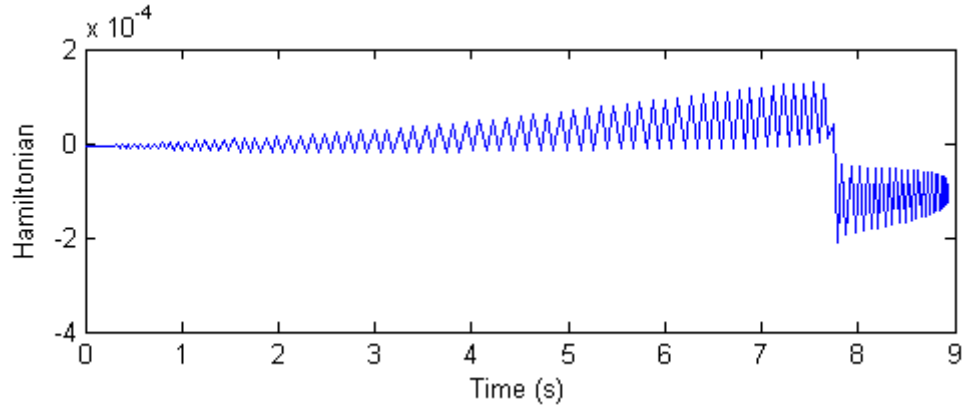


Figure 52. Minimum energy solution (GPOPS): history of the Hamiltonian.

The solution GPOPS arrives at is feasible in that it does not violate any of the constraints presented in the problem formulation. Furthermore, the control history follows the logic that was derived from Minimum Principle stated in Equation (23).

<i>Costate</i>	<i>Initial Condition</i>	<i>Costate</i>	<i>Initial Condition</i>	<i>Costate</i>	<i>Initial Condition</i>
λ_1	-0.0335	λ_8	0.0225	λ_{15}	-0.1798
λ_2	0.0602	λ_9	0.1005	λ_{16}	0.0457
λ_3	-0.0356	λ_{10}	0.0460	λ_{17}	-0.2236
λ_4	-0.1013	λ_{11}	-0.1932	λ_{18}	-0.0086
λ_5	-0.2672	λ_{12}	-0.1462	λ_{19}	-0.1366
λ_6	-0.0714	λ_{13}	0.1229	λ_{20}	-0.3664
λ_7	-0.0460	λ_{14}	-0.1501	t_f	8.8943

Table 12. The initial values of costates and t_f as defined by GPOPS for minimum energy (quadratic-control) solution.

3. Minimum Absolute Control (Minimum Fuel)

To avoid using an absolute value (which is a nondifferentiable function at the value zero), the running cost in Equation (62) was substituted with:

$$f_0 = (u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9 + u_{10} + u_{11} + u_{12}) \quad (93)$$

with the corresponding modifications in the control allocation:

$$\mathbf{u} = \left[\frac{u_x^+}{u_{x\max}}, \frac{u_y^+}{u_{y\max}}, \frac{u_z^+}{u_{z\max}}, \frac{T_x^+}{T_{x\max}}, \frac{T_y^+}{T_{y\max}}, \frac{T_z^+}{T_{z\max}}, \frac{u_x^-}{u_{x\max}}, \frac{u_y^-}{u_{y\max}}, \frac{u_z^-}{u_{z\max}}, \frac{T_x^-}{T_{x\max}}, \frac{T_y^-}{T_{y\max}}, \frac{T_z^-}{T_{z\max}} \right]^T \quad (94)$$

so that now $\mathbf{0} \leq \mathbf{u} \leq \mathbf{1}$ (the superscript on the control, + or -, denotes the direction in the body axis). This also results in the following control relations (compare with Equation (21)):

$$\begin{aligned} f_x &= (q_4^{C2} + q_1^{C2} - q_2^{C2} - q_3^{C2})(u_1 - u_7) + 2(q_1^C q_2^C - q_3^C q_4^C)(u_2 - u_8) + 2(q_1^C q_3^C + q_2^C q_4^C)(u_3 - u_9) \\ f_y &= 2(q_1^C q_2^C + q_3^C q_4^C)(u_1 - u_7) + (q_4^{C2} - q_1^{C2} + q_2^{C2} - q_3^{C2})(u_2 - u_8) + 2(q_2^C q_3^C - q_1^C q_4^C)(u_3 - u_9) \\ f_z &= 2(q_1^C q_3^C - q_2^C q_4^C)(u_1 - u_7) + 2(q_2^C q_3^C + q_1^C q_4^C)(u_2 - u_8) + (q_4^{C2} - q_1^{C2} - q_2^{C2} + q_3^{C2})(u_3 - u_9). \end{aligned} \quad (95)$$

$$\begin{aligned} T_x &= (u_4 - u_{10}) \\ T_y &= (u_5 - u_{11}) \\ T_z &= (u_6 - u_{12}) \end{aligned}$$

Analyzing the structure of the switching function,

$$H^*(\lambda, \mathbf{x}, \mathbf{u}) = \sum_{i=1}^{12} u_i + \frac{1}{m} (\lambda_4 f_1 + \lambda_5 f_2 + \lambda_6 f_3) + \frac{1}{I_{11}^C} \lambda_7 u_4 + \frac{1}{I_{22}^C} \lambda_8 u_5 + \frac{1}{I_{33}^C} \lambda_9 u_6 \quad (96)$$

and comparing it with that of Equation (26), we can derive the optimal control strategy as follows:

$$\begin{aligned} u_i &= \begin{cases} 1, & \text{switch}_i < -m \\ 0, & \text{switch}_i \geq -m \end{cases} \quad i = 1, 2, 3 \\ \text{and } u_i &= \begin{cases} 1, & \lambda_{i+3} < -I_{jj}^C \\ 0, & \lambda_{i+3} \geq -I_{jj}^C \end{cases} \quad i = 4, 5, 6, j = i - 3 \\ \\ u_i &= \begin{cases} 0, & \text{switch}_{i-6} \leq m \\ 1, & \text{switch}_{i-6} > m \end{cases} \quad i = 7, 8, 9 \\ \text{and } u_i &= \begin{cases} 0, & \lambda_{i-3} \leq I_{jj}^C \\ 1, & \lambda_{i-3} > I_{jj}^C \end{cases} \quad i = 10, 11, 12, j = i - 9 \end{aligned} \quad (97)$$

Here, switch_i , for $i = 1, 2, 3$, are the same as in Equation (68). Hence in this case we can expect bang-off-bang control.

The results are presented in Figures 53–62 and Table 13. The value of the PI was computed as $J = 2.4851$. In Figure 20, the controls u_i , $i = 1, \dots, 6$, acting in the positive direction along the corresponding body axis appear as positive values, and u_i , $i = 7, \dots, 12$ are shown with the negative sign. Note, that according to the optimal control structure of Equation (97) the switching occur when the switches exceed a certain value, not zero as for example in Equation (69). For instance, for u_z the motion starts with some short control input in the negative direction, corresponding to u_9^- , and ends with some short control input in the positive direction, corresponding to u_3^+ (during the remaining time the control u_z is zero). Also, the body mounted translational actuator, u_x , shows rapidly oscillating control values near the end of the maneuver, which is a property of a singular arc. The resulting rapidly oscillating control in that region would make the solution incredibly difficult to implement.

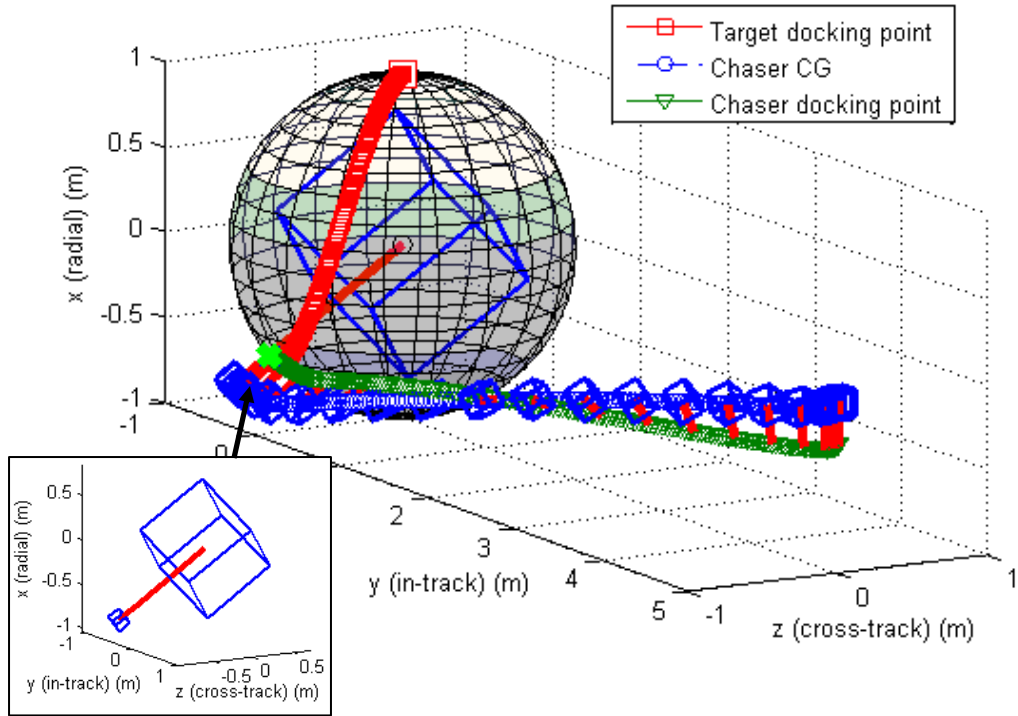


Figure 53. Minimum fuel solution (GPOPS): 3D plot of optimal rendezvous trajectory.

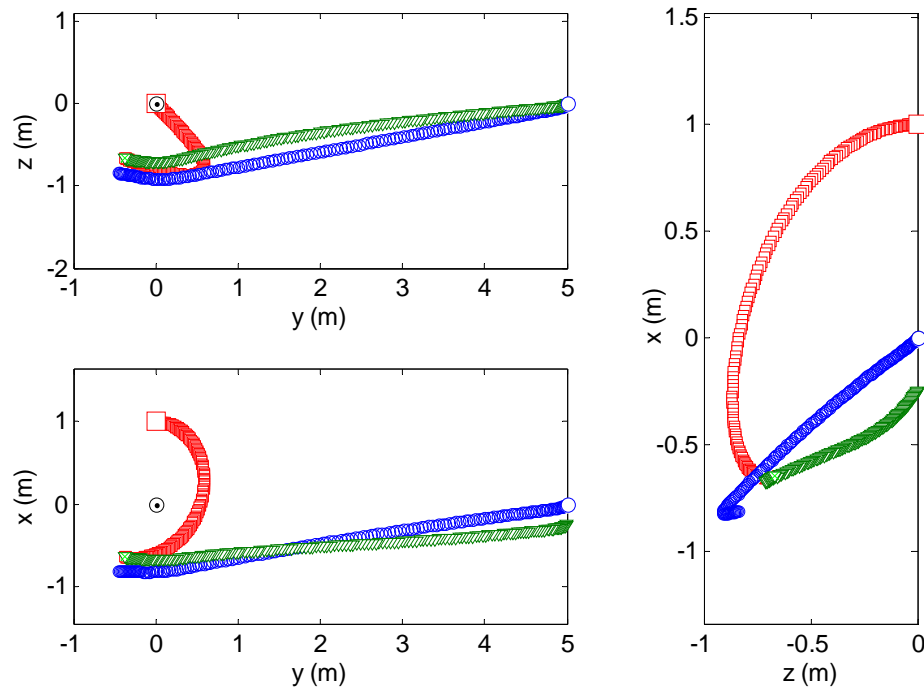


Figure 54. Minimum fuel solution (GPOPS): 2D plots of optimal rendezvous trajectory.

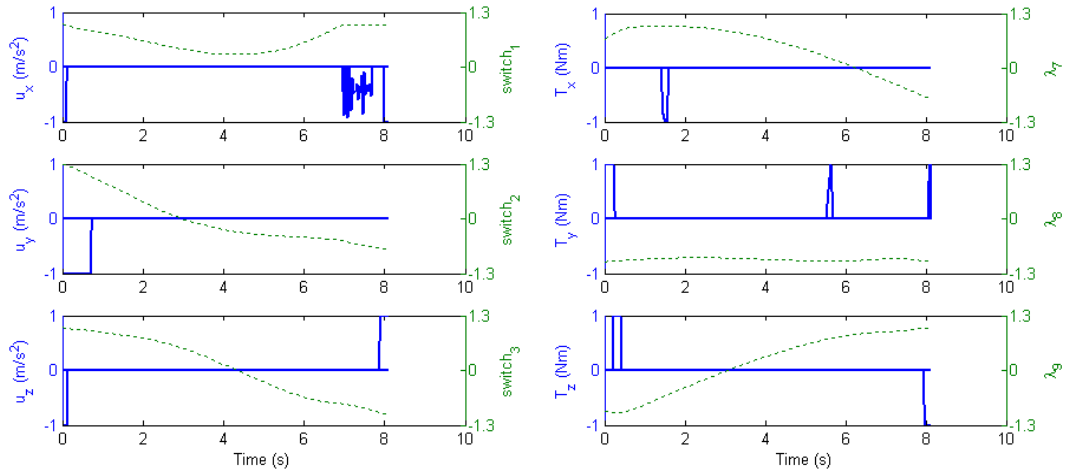


Figure 55. Minimum fuel solution (GPOPS): control history with respect to the optimal trajectory.

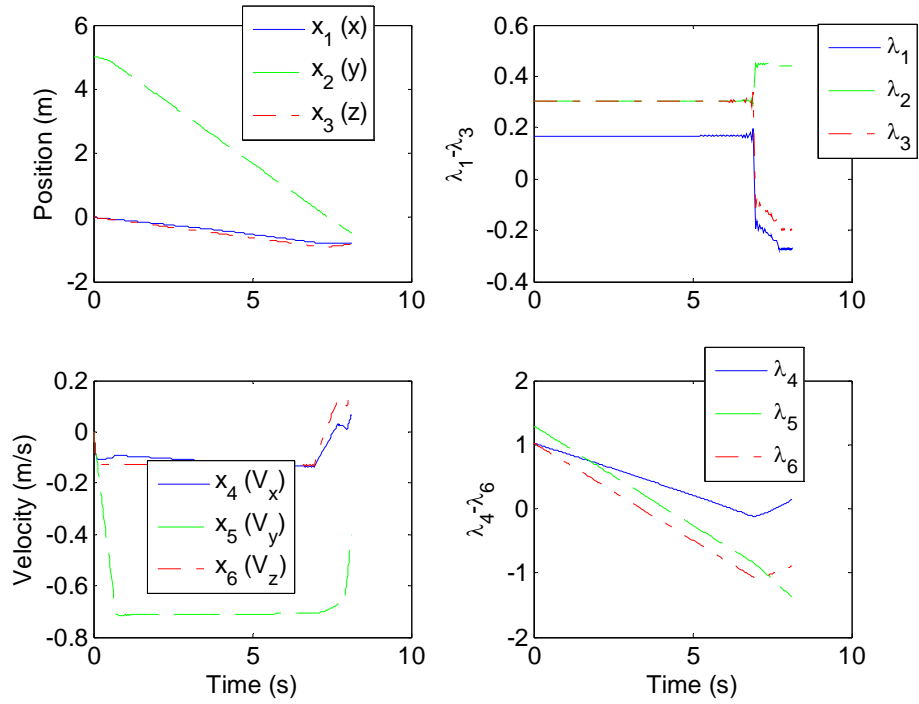


Figure 56. Minimum fuel solution (GPOPS): state and costate histories for the translational variables of the chaser vehicle.

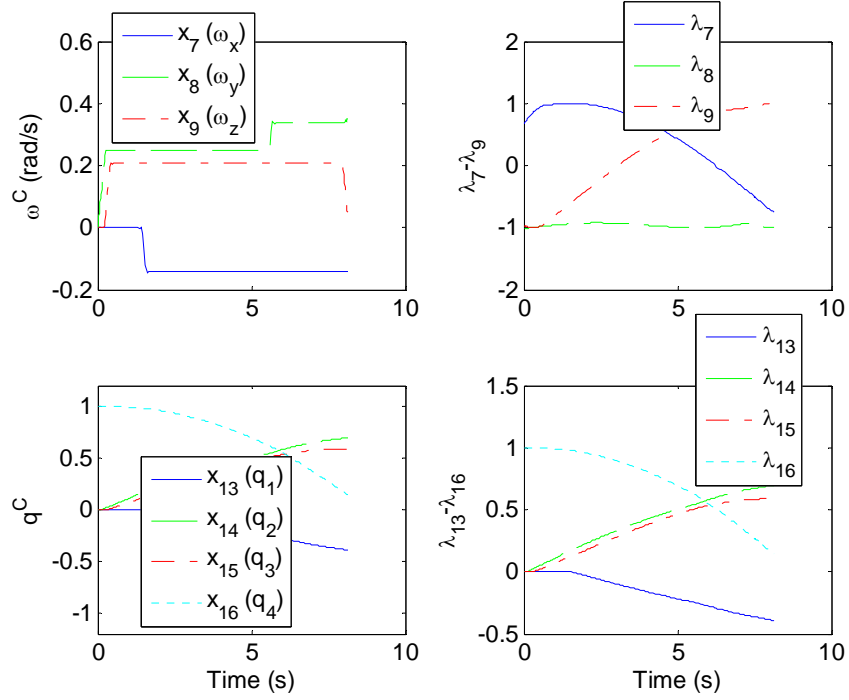


Figure 57. Minimum fuel solution (GPOPS): state and costate histories for the defining angular parameters of the chaser vehicle.

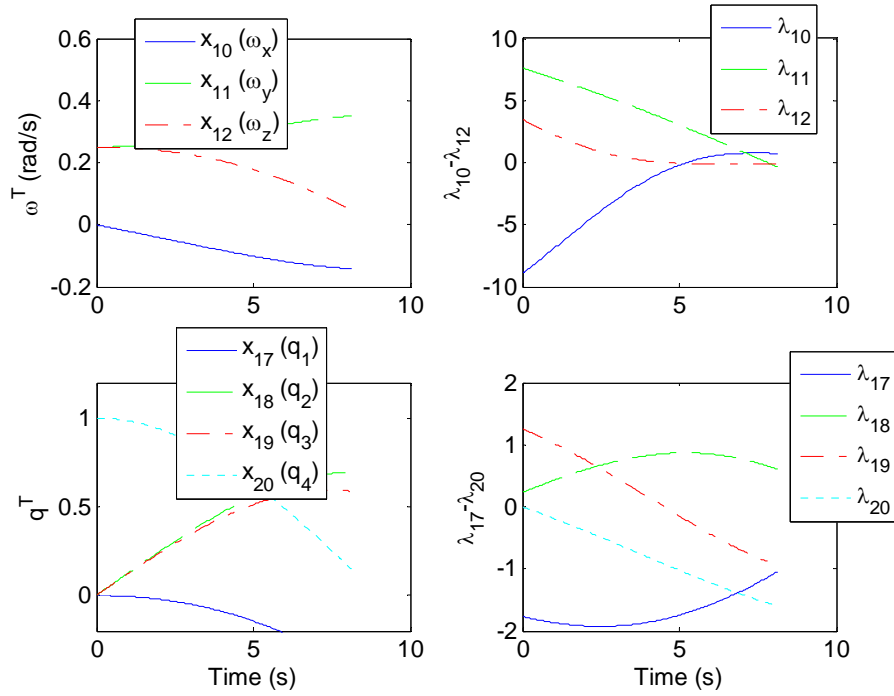


Figure 58. Minimum fuel solution (GPOPS): state and costate histories for the RSO.

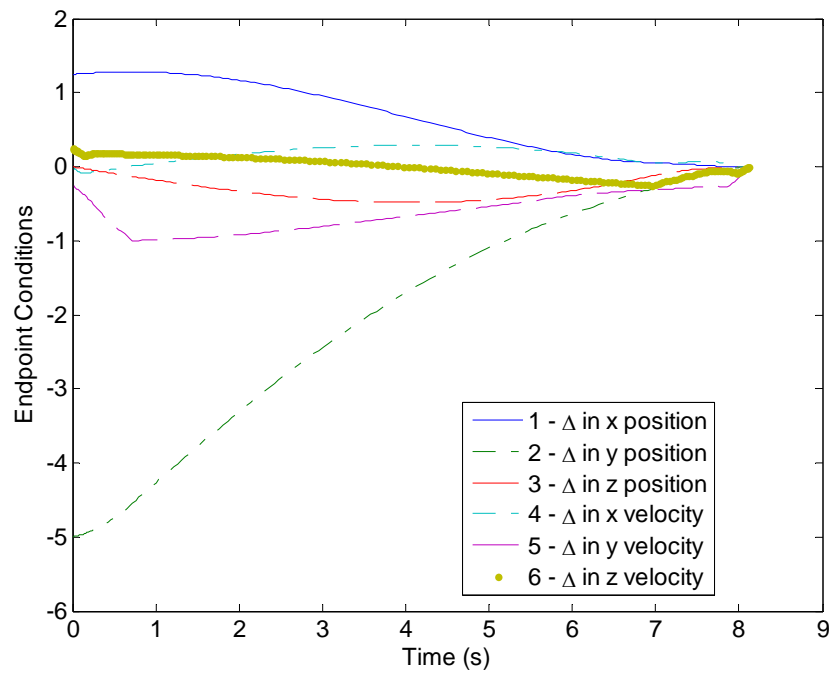


Figure 59. Minimum fuel solution (GPOPS): history of the translational endpoint discrepancies.

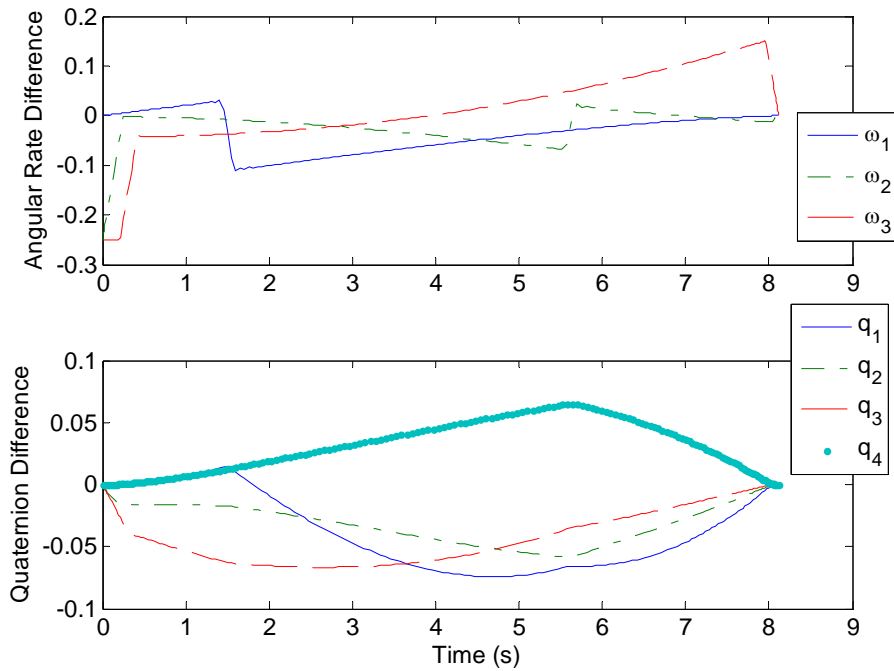


Figure 60. Minimum fuel solution (GPOPS): history of the rotational endpoint discrepancies.

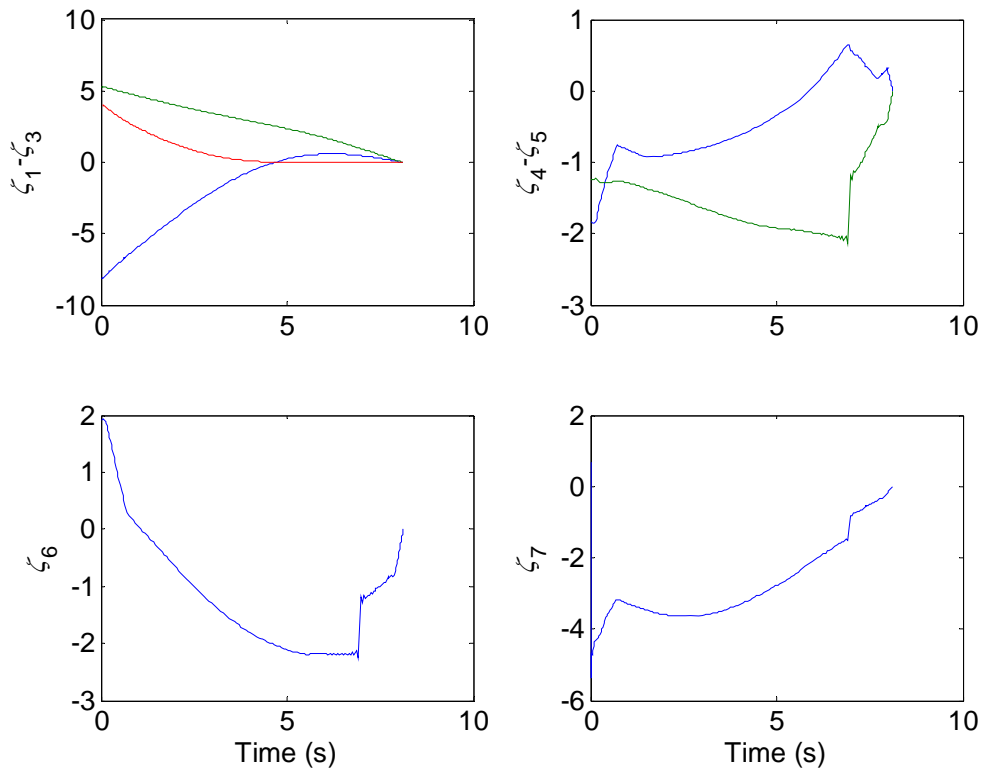


Figure 61. Minimum fuel solution (GPOPS): history of the transversality conditions.

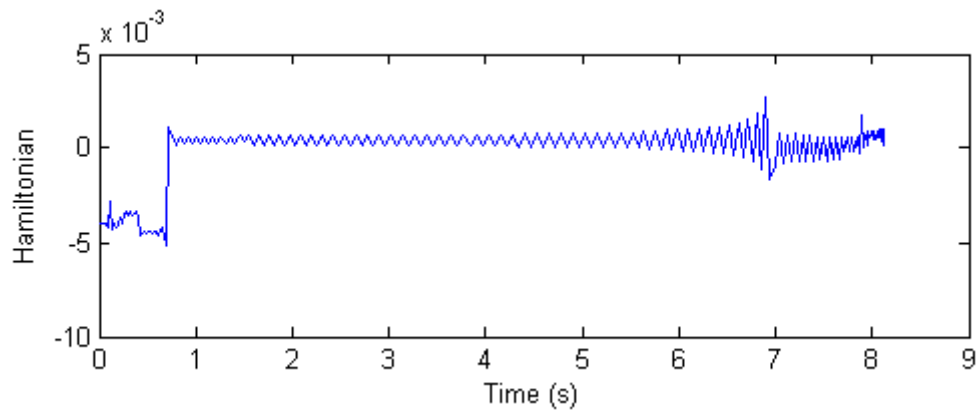


Figure 62. Minimum fuel solution (GPOPS): history of the Hamiltonian.

<i>Costate</i>	<i>Initial Condition</i>	<i>Costate</i>	<i>Initial Condition</i>	<i>Costate</i>	<i>Initial Condition</i>
λ_1	-0.2725	λ_8	-1.0009	λ_{15}	1.6243
λ_2	0.4399	λ_9	1.0180	λ_{16}	0.9074
λ_3	-0.1960	λ_{10}	0.7419	λ_{17}	1.1494
λ_4	0.1341	λ_{11}	-0.3327	λ_{18}	-3.2616
λ_5	-1.3660	λ_{12}	-0.1168	λ_{19}	-4.2237
λ_6	-0.8983	λ_{13}	-1.5419	λ_{20}	-2.4355
λ_7	-0.7419	λ_{14}	2.6216	t_f	8.1182

Table 13. The initial values of costates and t_f as defined by GPOPS for minimum fuel solution.

4. Solution Comparisons and Propagation

a. Solution and Comparison

Table 14 summarizes the results for each 200-node solution presented in Chapter III.C. Obviously, the scenario with the minimum time PI uses the most fuel to complete the maneuver. The minimum energy solution has an order of magnitude lower cost, as it spreads the control over the entire maneuver. It also exhibits a smoothest controls time histories, but maneuver takes the longest to complete. The minimum fuel solution results in about the same duration of the maneuver as the minimum energy solution, but uses about 30% less fuel. As opposed to minimum energy, it exhibits a bang-off-bang control structure that was expected with this type of the running cost.

	<i>Minimum time</i>	<i>Minimum-Quad-Control</i>	<i>Minimum fuel</i>
<i>Time of the maneuver</i>	3.5086	8.8943	8.1182
<i>Energy expenditure</i>	10.4471	0.2445	1.1587
<i>Fuel expenditure</i>	20.9419	3.6941	2.4863
<i>Computation time</i> * (sec)	33,370	86,040	84,261

Table 14. Comparison of solutions for the three optimal control problems.

Table 14 also presents the computational time it took to converge to a solution from the very same initial guess. All computations were performed on a 2.33GHz Dell Precision M90 desktop computer with an Intel T7600 processor and 1Gb of RAM. As compared to the results discussed in Chapter II (Boyarko et al. 2009a), the required central processing unit (CPU) time is about three times longer. This is due to the fact that 200 nodes were used in the simulation as compared to 150 nodes to that of

Chapter II. As explained by Yakimenko (Yakimenko, Xu and Basset 2008), the required CPU time can be drastically decreased by decreasing the number of nodes, but apparently the lesser-node solution would probably not catch short impulses of Figure 47.

The ultimate goal is to allow trajectory generation to be carried in real time onboard spacecraft. At this point, this cannot be realized using either pseudospectral methods (Yakimenko et al. 2008).

b. Propagating the Solution

The goal of this section is to check the feasibility of using the optimal solution in order to control the system in feed-forward mode. Consider the very first case, when the minimum time solution was obtained. Now that we have the controls, let us integrate the equation of motion derived in Chapter III.A with a time step of 0.0001 seconds and use the calculated controls as an input. The resulting trajectory is shown in Figures 63 and 64. The time history of the differences in chaser and RSO docking position, velocity, angular position and angular rate are shown in Figure 65. Table 15 lists the resulting state variable deviations from the desired endpoints stated in Equations (55)-(58) at the final time.

The propagated trajectory appears to be very close to that of Figures 32 and 32, i.e., propagation of the dynamics using the minimal time control output from GPOPS results in a feasible trajectory that is similar to the resulting trajectory generated by GPOPS. This means that if the solution could be obtained in real time, with a perfect model of the real system, it can be passed to the control system by feed-forwarding the controls time histories.

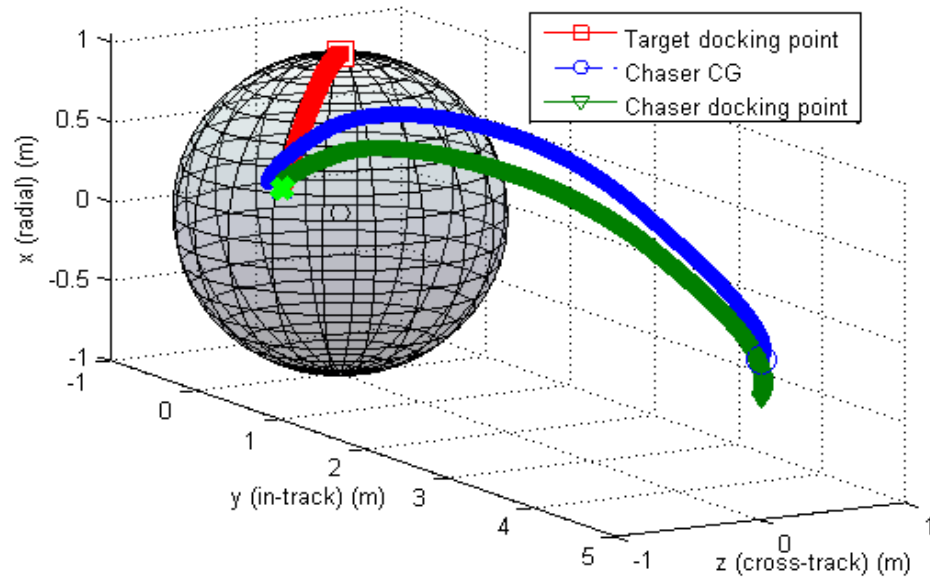


Figure 63. Propagated trajectory using the minimum time control history from GPOPS.

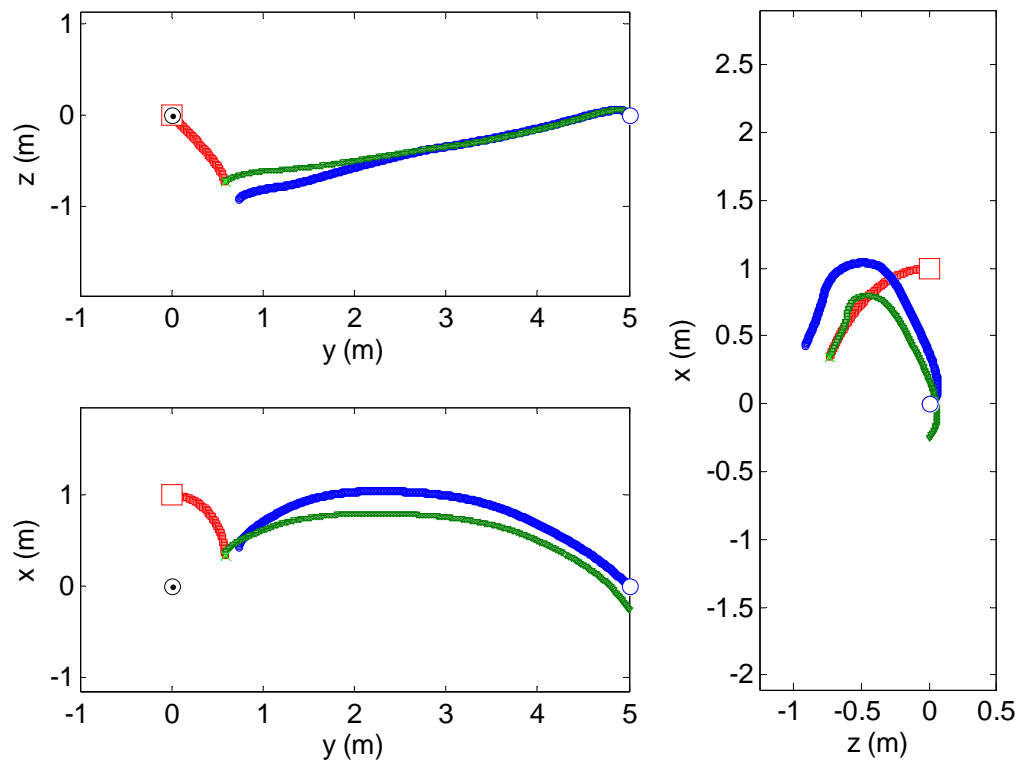


Figure 64. Propagated trajectory using the minimum time control history.

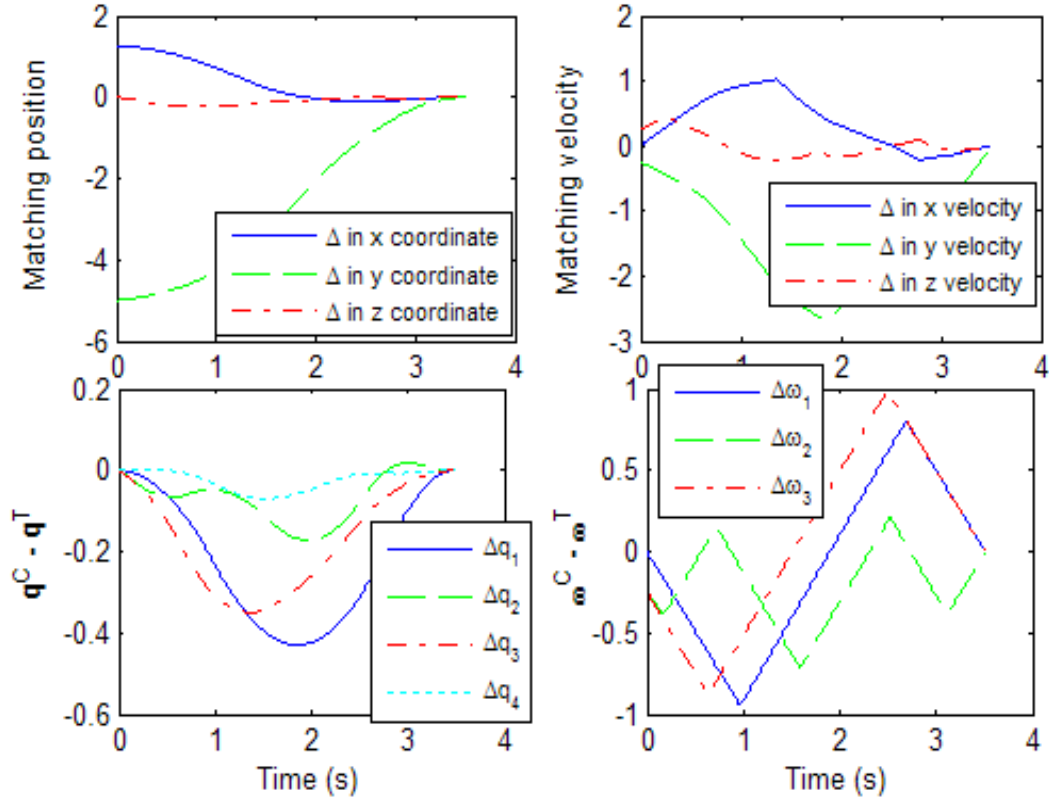


Figure 65. Endpoint conditions on chaser and RSO docking position, velocity, angular position and angular rate.

e	<i>Resulting Value</i>	e	<i>Resulting Value</i>	e	<i>Resulting Value</i>
$e_1(t_f)$	-0.0003	$e_8(t_f)$	0.0001	$\zeta_2(t_f)$	N/A
$e_2(t_f)$	-0.0021	$e_9(t_f)$	-0.0001	$\zeta_3(t_f)$	N/A
$e_3(t_f)$	-0.0007	$e_{10}(t_f)$	0.0005	$\zeta_4(t_f)$	N/A
$e_4(t_f)$	-0.0003	$e_{11}(t_f)$	0.0004	$\zeta_5(t_f)$	N/A
$e_5(t_f)$	0.0015	$e_{12}(t_f)$	-0.0002	$\zeta_6(t_f)$	N/A
$e_6(t_f)$	-0.0014	$e_{13}(t_f)$	-0.0000	$\zeta_7(t_f)$	N/A
$e_7(t_f)$	-0.0001	$\zeta_1(t_f)$	N/A	$H(t_f)$	N/A

Table 15. Value of terminal conditions at the final time.

IV. TIME-OPTIMAL REORIENTATION OF A SPACECRAFT USING A DIRECT OPTIMIZATION METHOD BASED ON QUATERNION REPRESENTATION OF THE INVERSE DYNAMICS

This chapter of the manuscript focuses on using the equations of motion to define the controls for a given attitude trajectory. This is the first step in defining the necessary equations and relations for the IDVD method for a close approach that would enable inspection or docking. In later sections, the methodology will be applied to the translational portion of the problem, but first the attitude control is considered. An initial trajectory is provided as an initial guess. The trajectory is then perturbed in the solution space by varying higher order derivatives on the endpoints until an acceptable solution is found. It should be noted that using inverse dynamics to optimize a rotational motion of a satellite has been already evaluated by other authors as well (Louembet, Cazaurang, Zolghardi, Charbonnel, and Pittet 2007; and McInnes 1998). However, the Euler angles, suffering from well-known kinematic singularities were used, and no attempt to decouple the domains of space and time were made. Furthermore, this research extends the situations to more realistic scenarios where nonzero rates and accelerations are present as in cases of target tracking and docking.

A. SPACECRAFT MODEL AND ATTITUDE TRAJECTORY OPTIMIZATION PROBLEM

The first step is to define the dynamics of the system in question, a reorienting spacecraft. The rotational dynamics of the spacecraft can be described by Euler's rotational equations of motion. Written in the body-fixed principal axis, this results in the vector equation (Greenwood 1987; Wie 1988):

$$\mathbf{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} = \mathbf{T}, \quad (98)$$

which expands into the scalar equations:

$$\begin{aligned}
\dot{\omega}_x = \alpha_x &= \frac{(I_{22} - I_{33})\omega_y\omega_z + T_x}{I_{11}}, \\
\dot{\omega}_y = \alpha_y &= \frac{(I_{33} - I_{11})\omega_x\omega_z + T_y}{I_{22}}, \\
\dot{\omega}_z = \alpha_z &= \frac{(I_{11} - I_{22})\omega_y\omega_x + T_z}{I_{33}}.
\end{aligned} \tag{99}$$

In turn, rotational kinematics can be described using quaternions, $\mathbf{q} = [q_1, q_2, q_3, q_4]^T$. The definition of the quaternion is taken from Equation (8) where σ is the scalar value of the rotational displacement about the eigenaxis and \mathbf{p} is unit vector that describes the direction of the eigenaxis. The dynamics of the unit quaternion are described as (Greenwood 1987; Wie 1988):

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix}. \tag{100}$$

In Equations (99) and (100), $\mathbf{I} = \text{diag}([I_{11}, I_{22}, I_{33}])$ is the inertia matrix (along the principal axes), $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T$ is the vector of angular velocities, and $\mathbf{T} = [T_x, T_y, T_z]^T$ is the vector torques (bounded controls).

The problem in question is to find the slew trajectory (quaternion time history) for a satellite subject to specific constraints that minimizes the time to complete the maneuver, t_f . This is expressed as minimizing the PI:

$$J = \int_0^{t_f} dt, \tag{101}$$

while reorienting a satellite from the initial conditions $\boldsymbol{\omega}_0, \mathbf{q}_0$ to final conditions $\boldsymbol{\omega}_f, \mathbf{q}_f$ for a system (99)-(100), subject to constraints on controls

$$\mathbf{T}_{\min} \leq \mathbf{T} \leq \mathbf{T}_{\max}. \tag{102}$$

Bilimoria and Wie (1993) formulated this problem for a rest-to-rest maneuver and presented the solution using an indirect method. They showed that, in general, for the case of symmetric body with bounds on each torque component, it does not result in an eigenaxis maneuver. In addition to that, the following section presents even a more general solution obtained off-line to be used along with that of Bilimoria and Wie (1993) as a reference one for the proposed on-line solution obtained using a direct method exploiting inverse dynamics of Equations (99)–(100).

Table 1 shows the different test cases examined in this chapter. Test Case 1 was taken directly from Bilimoria and Wie (1993), while the others were chosen to illustrate different scenarios of interest.

Case	<i>Normalized Inertia Matrix</i>
<i>Case 1</i>	$\mathbf{I} = \text{diag}([1, 1, 1])$
<i>Case 2</i>	$\mathbf{I} = \text{diag}([3, 1, 2])$

Table 16. Description of the test cases.

This chapter considers two basic scenarios assuming a $\phi = 90^\circ$ and $\phi = 180^\circ$ slew maneuver about the z -axis (so that $\mathbf{q}_0 = [0, 0, 0, 1]^T$ and $\mathbf{q}_f = [0, 0, \sin \frac{1}{2}\phi, \cos \frac{1}{2}\phi]^T$) with zero and nonzero normalized body rates at endpoints: a) $\boldsymbol{\omega}_0 = \boldsymbol{\omega}_f = \mathbf{0}_{3 \times 1}$, b) $\boldsymbol{\omega}_0 = -\boldsymbol{\omega}_f = \frac{1}{10}\mathbf{1}_{3 \times 1}$, and c) $\boldsymbol{\omega}_0 = -\boldsymbol{\omega}_f = \mathbf{1}_{3 \times 1}$. For the normalized states, the constraints (5) take the form $-\mathbf{1}_{3 \times 1} \leq \mathbf{T} \leq \mathbf{1}_{3 \times 1}$.

B. SOLVING THE PROBLEM USING THE GAUSS PSEUDOSPECTRAL METHOD

First, the problem formulated in Chapter IV.A is addressed using the same pseudospectral direct method as in Chapters II and III. The goal is to have some reference solutions for comparison to the solutions obtained through other methods, specifically IDVD.

The optimal (Bilimoria and Wie 1993) solution has been matched using a different optimization method in previous work (Fleming 2004).

Figures 66–68 show the results of applying GPOPS to obtain minimum time solutions for a 180° slew of a satellite. Specifically, Figures 66 and 67 present time histories of all states and controls for the solution that involves 100 nodes, which results in $(100-2) \times 10 = 980$ variable parameters, derived from the 98 internal node points for each of the 7 state and 3 control variables. Figure 68 depicts the 3D representation of the solution in inertial space, clearly showing that it is not an eigenaxis maneuver, with an inclination of the z_b axis during rotation in the $x_b y_b$ direction.

This solution compares with the solution presented in Bilimoria and Wie (1993) fairly well. The final calculated maneuver time, t_f , was found to be 3.243 seconds. However, it took almost two hours of CPU time to obtain this solution. Another observation is that because of the nature of the system described by Equation (99), the optimal control has a bang-bang structure as shown in Figure 67. That results in the maximum magnitude of the angular acceleration at the boundary points (for Case 1 angular accelerations are simply equal to the corresponding controls). It means arriving at the terminal conditions with the maximum angular acceleration. Also, if we have to update a trajectory while the satellite performs this rotation (to accommodate possible disturbances and unmodeled dynamics), it would cause discontinuities of angular acceleration (sudden jumps in controls). Increasing the order of the system to account for the boundary conditions on angular accelerations will obviously cause a slight degradation of the PI and a further increase of the required CPU time to obtain a solution. Hence, although in this case GPOPS does produce a valid solution, as previously stated in Chapters II and III, it is not practical and currently cannot be used for online computations.

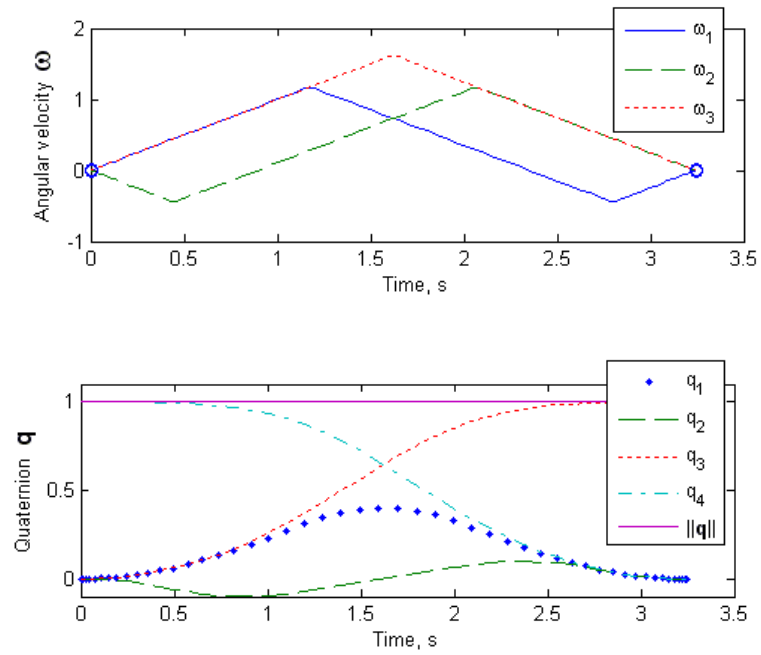


Figure 66. Case 1 (GPOPS solution): time histories of the state variables.

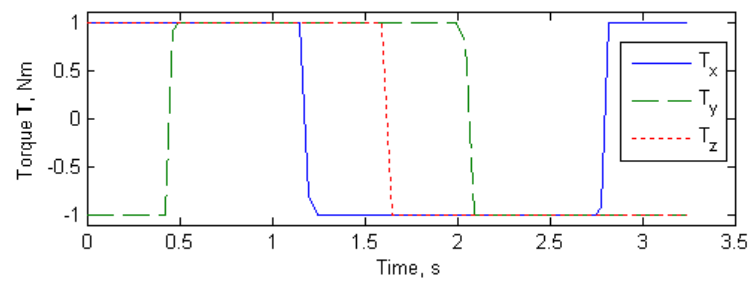


Figure 67. Case 1 (GPOPS solution): time history of the controlling torques.

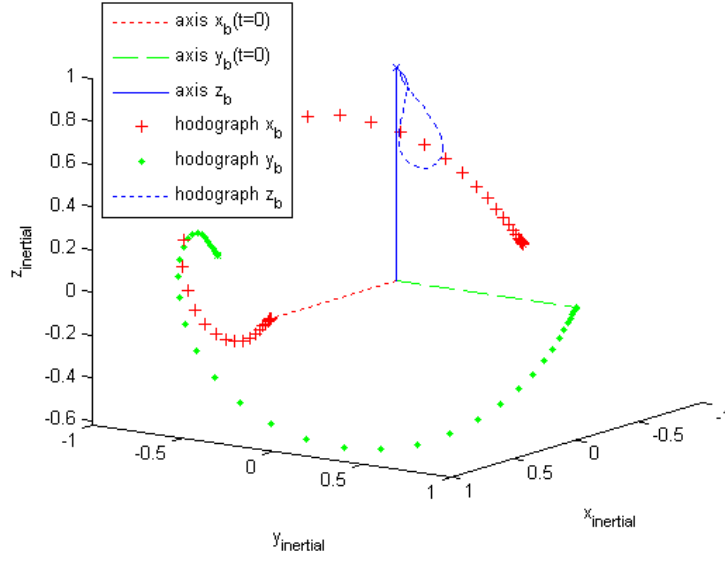


Figure 68. Case 1 (GPOPS solution): the 3D representation of the solution.

As pointed out in GPOPS documentation (Rao et al. 2009), reducing the number of nodes may lead to a more robust (in terms of computational time) result, therefore an attempt was made to obtain a solution of the same problem using a lesser number of nodes. These GPOPS solutions are shown in Figures 69–71.

It turns out that for a lesser number of nodes the GPOPS converges to different solutions. To this end, Figure 68 shows time histories of the angular velocity components for the 25- and 50-node solution (involving 230 and 480 varied parameters, respectively). Obviously, they are different from that of the 100-node solution in Figure 64. While a 25-node solution is simply symmetrical with respect to the 100-node solution, as can be seen by comparing Figure 66 and Figure 68 showing an inclination of the z_b axis during rotation in the $-x_b y_b$ direction, and represents another equally optimal solution out of possible four solutions (Fleming 2004). This is due to the equivalent possibilities of positive or negative inclination in the z_b direction coupled with possibilities of $-x_b y_b$ and $x_b y_b$ directional rotation for the 180° maneuver about a principal axis. The

possibility of the positive or negative inclination in the z_b direction also leads to two equivalent solutions for similar slews less than 180° . A 50-node solution, shown in Figure 69, appears not to be valid (optimal) at all.

As expected, decreasing the number of nodes leads to a substantial decrease in the computational time, but as shown above the method could produce a nonvalid solution. Also, even if it produces a valid solution the time histories for control torques, Figure 70, may not be trackable by the inner-loop controllers. Therefore, the solutions obtained by GPOPS may not be used in a real time feed-forward control scheme.

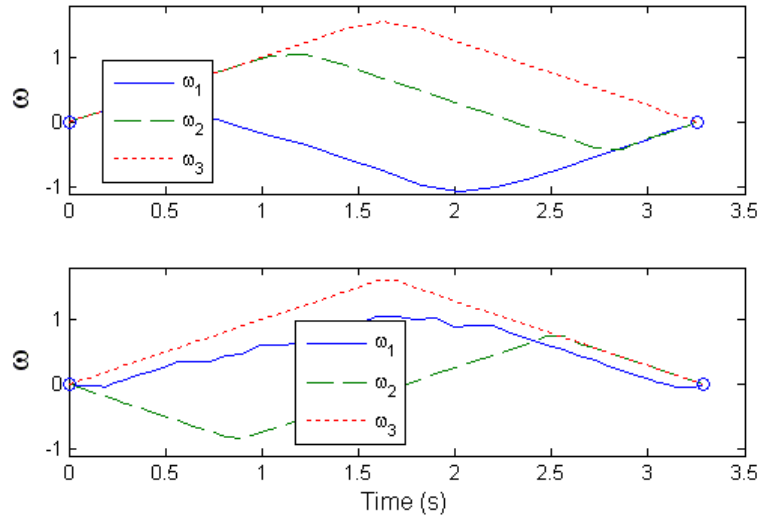


Figure 69. Case 1 (GPOPS solution): comparison of time histories of angular velocity components obtained for 25 nodes (top) and 50 nodes (bottom).

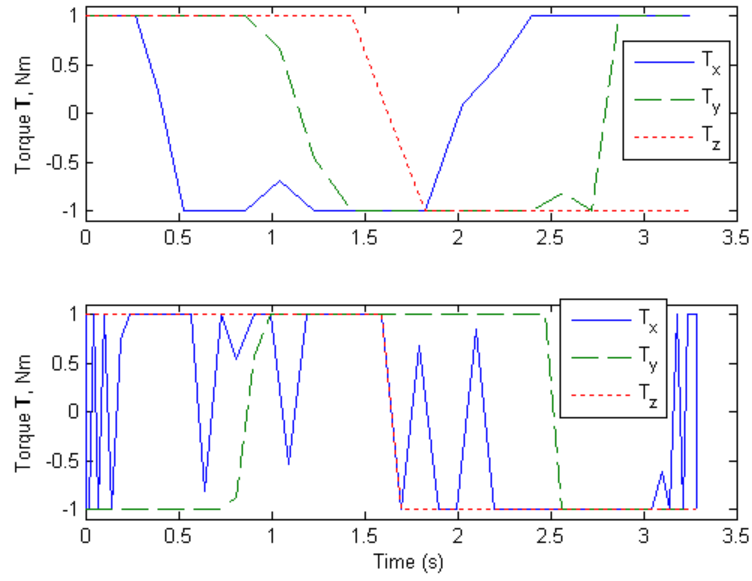


Figure 70. Case 1 (GPOPS solution): comparison of time histories of torques, obtained for 25 nodes (top) and 50 nodes (bottom).

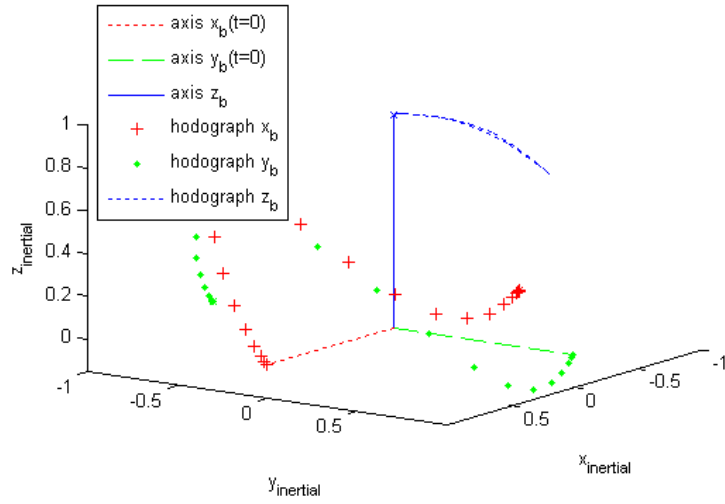


Figure 71. Case 1 (GPOPS solution): the 3D representation of the 25-node solution.

For Case 2, the nonsymmetrical inertia with the bounds on individual control torques, the solution is slightly different (Figures 72–73). The overall characteristic of this and other solutions involving different sets of the boundary conditions will be

presented in Chapter IV.E, but the general tendency is the same—it requires at least a hundred nodes to produce a valid and feasible off-line solution. Yet, GPOPS again presents a good and easy-to-use tool to produce reference trajectories that can be used for comparison with solutions obtained using other approaches.

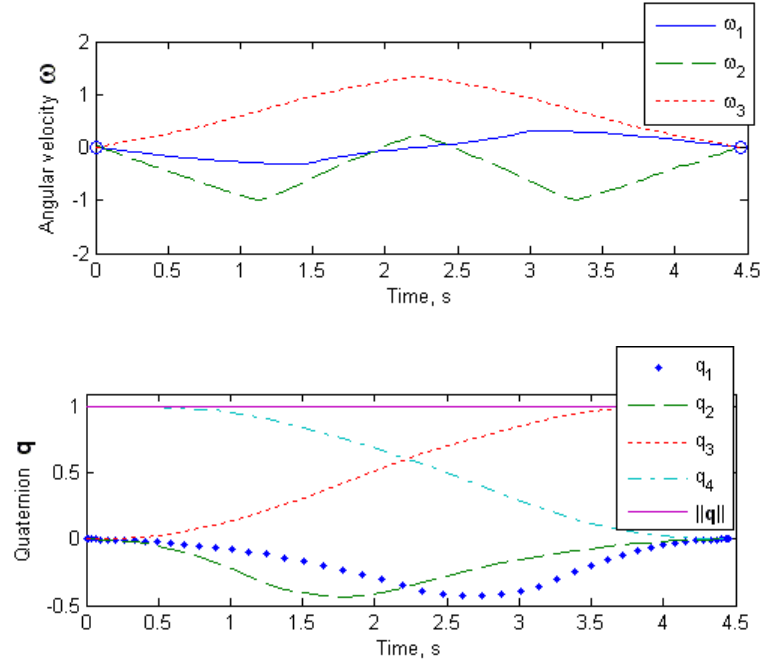


Figure 72. Case 2 (GPOPS solution): time histories of the state variables.

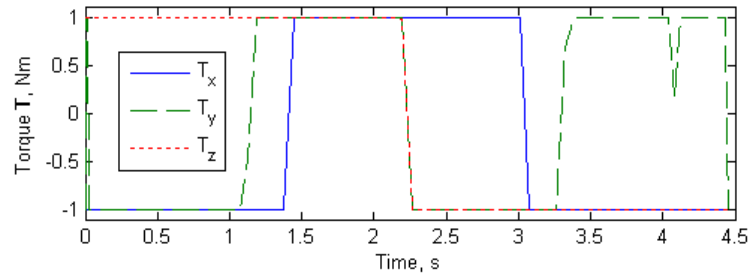


Figure 73. Case 2 (GPOPS solution): time history of the controlling torques.

C. APPLICATION OF INVERSE DYNAMICS IN THE VIRTUAL DOMAIN APPROACH WITH QUATERNION ATTITUDE REPRESENTATION

One of the two main ideas of Inverse Dynamics in the Virtual Domain (IDVD) method is exploiting the differential flatness property of the equations of motion (Yakimenko 2000; Cowling, Yakimenko, Whidborne and Cooke 2007; Yakimenko and Slegers 2009). In the above problem, this relates to the fact that all the state and control variables can be expressed as explicit functions of the output variable or time derivatives of the output variable, which in this case is the quaternion itself:

$$\boldsymbol{\omega} = f_1(\mathbf{q}, \dot{\mathbf{q}}), \quad \mathbf{T} = f_2(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}). \quad (103)$$

Another aspect of IDVD involves handling computations in the virtual domain allowing space and time decoupling. By doing so, a trajectory can be computed while also manipulating the speed at which that trajectory is followed. The following sections present a novel parameterization for the output variable, components of the quaternion, and develop a step-by-step computational scheme.

1. Quaternion Parameterization

In order to parameterize the problem, the output trajectory is approximated using some combination of basis functions. The standard approach would be to choose some combination of polynomials or trigonometric functions for the output variables (Yakimenko 2000; Cowling et al. 2007; Yakimenko and Slegers 2009). While this may be straightforward when dealing with state variables in translational space, it may become more challenging when dealing with expressions for orientation.

While a quaternion may be the preferred method to express attitude because of the lack of singularities, choosing basis functions becomes more challenging because a nonlinear unit norm condition needs to be preserved across the quaternion history (Milam 2003). Simply choosing arbitrary polynomials for the 1st three components of the quaternion and attempting to enforce the quaternion constraint is insufficient because it cannot be guaranteed the constraint will be met throughout the maneuver (specifically if one of the iterations on a polynomial provides a norm > 1 in any one of the components), so simply tacking on a penalty function for violating the constraint will not work. For

this reason, a specific polynomial expression for the quaternion was chosen inspired by the work of Kim, Kim and Shin (1995). This consists of expressing the quaternion time history as an exponential function containing a constant parameter multiplied by a Bezier polynomial of a degree n :

$$\mathbf{q}(\tau) = \mathbf{q}_0 \prod_{i=1}^n \exp(\tilde{\omega}_i \tilde{\beta}_{i,n}(\tau)), \quad (104)$$

where

$$\tilde{\beta}_{i,n}(\tau) = \sum_{j=i}^n \beta_{j,n}(\tau) \quad (105)$$

and

$$\beta_{i,n}(\tau) = \binom{n}{i} (1-\tau)^{n-i} \tau^i. \quad (106)$$

A complete list of quaternion properties employed in this dissertation is stated in the Appendix. Note, that in Equations (104)–(106) $\tau \in [0;1]$ an abstract argument is used instead of time. This allows us to exploit certain attributes of the Bezier polynomials and define properties at the beginning and endpoints. For example, the analytic expressions of the 5th order Bezier Polynomial, $\tilde{\beta}_{i,5}(\tau)$, are as follows:

$$\begin{aligned} \tilde{\beta}_{1,5}(\tau) &= \tau^5 - 5\tau^4 + 10\tau^3 - 10\tau^2 + 5\tau, \\ \tilde{\beta}_{2,5}(\tau) &= -4\tau^5 + 15\tau^4 - 20\tau^3 + 10\tau^2, \\ \tilde{\beta}_{3,5}(\tau) &= 6\tau^5 - 15\tau^4 + 10\tau^3, \\ \tilde{\beta}_{4,5}(\tau) &= -4\tau^5 + 5\tau^4, \\ \tilde{\beta}_{5,5}(\tau) &= \tau^5. \end{aligned} \quad (107)$$

In this case,

$$\mathbf{q}(\tau) = \mathbf{q}_0 \exp(\tilde{\omega}_1 \tilde{\beta}_{1,5}(\tau)) \exp(\tilde{\omega}_2 \tilde{\beta}_{2,5}(\tau)) \exp(\tilde{\omega}_3 \tilde{\beta}_{3,5}(\tau)) \exp(\tilde{\omega}_4 \tilde{\beta}_{4,5}(\tau)) \exp(\tilde{\omega}_5 \tilde{\beta}_{5,5}(\tau)). \quad (108)$$

A plot of the 5th order polynomials is shown in Figure 74.

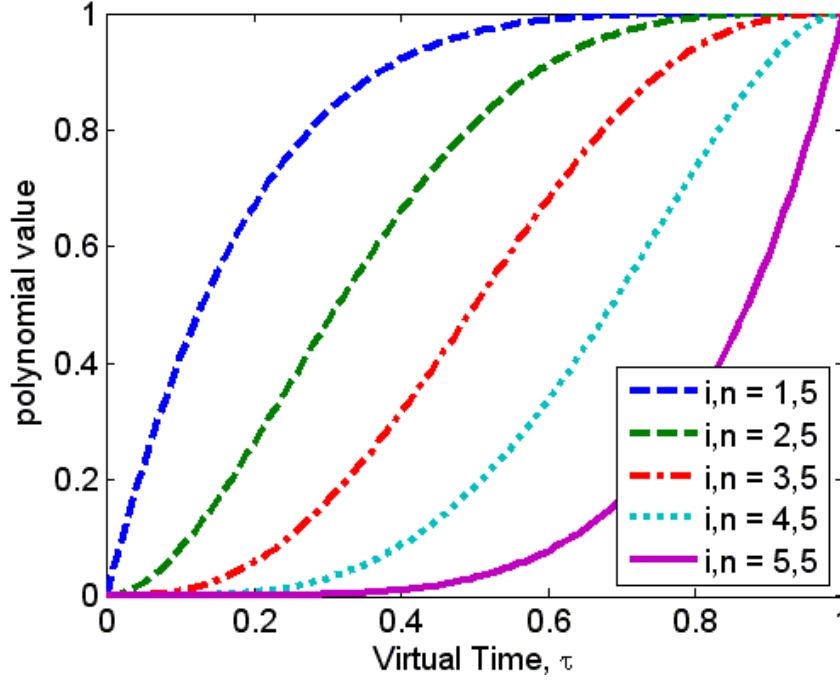


Figure 74. Value of 5th order Bezier Polynomials with respect to the virtual arc.

These expressions have the favorable properties:

$$\tilde{\beta}_{i,5}(0) = 0 \text{ and } \tilde{\beta}_{i,5}(1) = 1, \text{ for } i = 1, \dots, 5; \quad (109)$$

$$\begin{aligned} \dot{\tilde{\beta}}_{1,5}(0) &= \dot{\tilde{\beta}}_{5,5}(1) = 5, \quad \dot{\tilde{\beta}}_{1,5}(1) = \dot{\tilde{\beta}}_{5,5}(0) = 0, \\ \dot{\tilde{\beta}}_{i,5}(0) &= \dot{\tilde{\beta}}_{i,5}(1) = 0, \text{ for } i = 2, 3, 4; \end{aligned} \quad (110)$$

$$\begin{aligned} \ddot{\tilde{\beta}}_{1,5}(0) &= \ddot{\tilde{\beta}}_{5,5}(1) = 20, \quad \ddot{\tilde{\beta}}_{1,5}(1) = \ddot{\tilde{\beta}}_{5,5}(0) = 0, \\ \ddot{\tilde{\beta}}_{2,5}(0) &= \ddot{\tilde{\beta}}_{4,5}(1) = -20, \quad \ddot{\tilde{\beta}}_{2,5}(1) = \ddot{\tilde{\beta}}_{4,5}(0) = 0, \\ \ddot{\tilde{\beta}}_{3,5}(0) &= \ddot{\tilde{\beta}}_{3,5}(1) = 0, \end{aligned} \quad (111)$$

which fix the value of the polynomial and its derivatives at the endpoints specified by values of $\tau \in [0; \tau_f]$ where we set $\tau_f = 1$ that will act as an abstract virtual time argument to exploit the above properties in Equations (109)–(111). The values of $\tilde{\omega}_i$ are coefficients defined by the following relation:

$$\tilde{\omega}_i = \ln(\tilde{\mathbf{q}}_{i-1}^{-1} \tilde{\mathbf{q}}_i), \text{ for } i = 1, \dots, 5. \quad (112)$$

Here, $\tilde{\mathbf{q}}_i$ are the constant column vectors that act as control points and $\tilde{\boldsymbol{\omega}}_i$ represents a constant augmented angular velocity vector based on Equation (112).

The prevailing idea is that at $\mathbf{q}(\tau=0)=\tilde{\mathbf{q}}_0$ and $\mathbf{q}(\tau=1)=\tilde{\mathbf{q}}_5$, where $\tilde{\mathbf{q}}_0$ and $\tilde{\mathbf{q}}_5$ can be fixed such that $\tilde{\mathbf{q}}_0 \triangleq \mathbf{q}(t_0)$ and $\tilde{\mathbf{q}}_5 \triangleq \mathbf{q}(t_f)$. This also results in a straightforward calculation of higher order derivatives of the quaternion curve with respect to the virtual domain argument τ . Results for the first derivative were presented in Kim for a 3rd order Bezier polynomial (Kim et al. 1995). The results of the 2nd order derivative of a 5th order Bezier based quaternion is shown in Equation (113):

(113)

Differentials of quaternion trajectories based on varying order Bezier polynomials can be analytically calculated by proper application of the chain rule. The polynomial was expanded from a 3rd to a 5th order in order to fix the first and second derivative of the quaternion function at the endpoints. By applying Equations (105)–(109), therefore exploiting the property that certain terms $\tilde{\beta}_{i,5}$, $\dot{\tilde{\beta}}_{i,5}$ and $\ddot{\tilde{\beta}}_{i,5}$ are equal to zero, derivative values at the endpoints can be calculated by the simple expression:

$$\left. \frac{d\mathbf{q}}{d\tau} \right|_{\tau=0} = 5\tilde{\mathbf{q}}_0\tilde{\boldsymbol{\omega}}_1, \quad \left. \frac{d\mathbf{q}}{d\tau} \right|_{\tau=1} = 5\tilde{\mathbf{q}}_5\tilde{\boldsymbol{\omega}}_5. \quad (114)$$

Note, in Equation (114), the first derivative, how the parameter $\tilde{\boldsymbol{\omega}}_i$ is related to the angular velocity at the endpoints. In similar fashion,

$$\begin{aligned} \left. \frac{d^2\mathbf{q}}{d\tau^2} \right|_{\tau=0} &= -20\tilde{\mathbf{q}}_0\tilde{\boldsymbol{\omega}}_1 + 25\tilde{\mathbf{q}}_0\tilde{\boldsymbol{\omega}}_1^2 + 20\tilde{\mathbf{q}}_0\tilde{\boldsymbol{\omega}}_2, \\ \left. \frac{d^2\mathbf{q}}{d\tau^2} \right|_{\tau=1} &= -20\tilde{\mathbf{q}}_4\tilde{\boldsymbol{\omega}}_4\tilde{\mathbf{q}}_4^{-1}\tilde{\boldsymbol{\omega}}_5 + 20\tilde{\mathbf{q}}_5\tilde{\boldsymbol{\omega}}_5 + 25\tilde{\mathbf{q}}_5\tilde{\boldsymbol{\omega}}_5^2. \end{aligned} \quad (115)$$

2. Mapping from the Virtual Arc to the Time Domain

Now that the trajectory is set using a virtual domain, a mapping must be employed to convert this trajectory into a time dependent one. To do this, a speed factor, λ , is defined that maps the points on the trajectory from the virtual domain to the time domain, therefore defining the final time of the maneuver:

$$\lambda = \frac{d\tau}{dt}, \quad t_f = \int_0^{t_f} \frac{d\tau}{\lambda}. \quad (116)$$

The Speed factor can be a constant parameter that simply stretches or shrinks time evenly or it can take the shape of more complex function. For this application, $\lambda(\tau)$ is restricted to a function that contains a reduced number of varied parameters, developed

for this dissertation, that still allow for the speeding up and slowing down along a trajectory. The representation is shown in Equation (117).

$$\lambda(\tau) = \lambda_0 + A\tau^2 + (1-\tau)^2 B + (1-(1-\tau)^2)C + (1-\tau^2)D \quad (117)$$

Obviously, more complex structures of $\lambda(\tau)$ will provide more flexibility in the trajectory. One main idea of Equation (117) is to keep $\lambda(\tau)$ positive for all time, as $\lambda(\tau) < 0$ would imply time marching backwards and $\lambda(\tau) = 0$ signifies that time has stopped. Also, keeping the speed factor of a form such that an analytic integral to Equation (117) exists not only provides computational efficiency and an accurate integration to the minimum time PI but also provides a continuous mapping from the virtual domain to the time domain. Alternatively stated, a continuous control history is available, whose resolution does not suffer from a limited number of node points. This attribute is later revisited in Chapter VI.D.2.

Although a speed vector of the form Equation (117) does not allow matching the optimal minimum time solutions exactly, varying the parameters contained within $\lambda(\tau)$ (λ_0 , A , B , C , and D) still allows variation of the speed along the trajectory defined by Equation (104) to produce feasible and easy to track suboptimal solutions.

3. Inverting the Dynamics

As a result of the mapping from virtual to time domain, the expression for the differential of \mathbf{q} with respect to time is:

$$\dot{\mathbf{q}} = \frac{d\mathbf{q}}{dt} = \frac{d\mathbf{q}}{d\tau} \lambda, \quad \ddot{\mathbf{q}} = \frac{d^2\mathbf{q}}{dt^2} = \frac{d^2\mathbf{q}}{d\tau^2} \lambda^2 + \frac{d\mathbf{q}}{d\tau} \frac{d\lambda}{d\tau} \lambda. \quad (118)$$

Now, if the trajectory in the virtual domain, $\mathbf{q}(\tau)$, is specified, along with the speed trajectory, $\lambda(\tau)$, the resulting trajectory of $\mathbf{q}(t)$, as well as its higher order derivatives, can be analytically expressed and mapped to the time domain.

Inverting kinematic equations (100) and differentiating the result yields analytical expressions for the angular velocity and angular acceleration:

$$\begin{aligned}\boldsymbol{\omega}(t) &= 2\mathbf{q}^{-1}(t)\dot{\mathbf{q}}(t), \\ \boldsymbol{\alpha}(t) &= \dot{\mathbf{q}}^{-1}(t)2\ddot{\mathbf{q}}(t) - \dot{\mathbf{q}}(t)\boldsymbol{\omega}(t).\end{aligned}\tag{119}$$

The torque history needed to follow such a trajectory is calculated by inverting Equation (99):

$$\begin{aligned}T_x(t) &= \alpha_x(t) + (I_{33} - I_{22})\omega_y(t)\omega_z(t), \\ T_y(t) &= \alpha_y(t) + (I_{11} - I_{33})\omega_1(t)\omega_z(t), \\ T_z(t) &= \alpha_z(t) + (I_{22} - I_{11})\omega_y(t)\omega_x(t).\end{aligned}\tag{120}$$

4. Matching Endpoint Conditions

From the preceding equations, a quaternion history can be developed based on the Bezier polynomial that satisfies predefined beginning and ending quaternion values as well as setting the angular velocity and angular acceleration at the endpoints. The desired angular velocity and acceleration dictate the quaternion derivatives at the endpoints to be as follows:

$$\begin{aligned}\dot{\mathbf{q}}(t_0) &= \mathbf{q}(t_0)\boldsymbol{\omega}(t_0), \\ \dot{\mathbf{q}}(t_f) &= \mathbf{q}(t_f)\boldsymbol{\omega}(t_f), \\ \ddot{\mathbf{q}}(t_0) &= \dot{\mathbf{q}}(t_0)2\boldsymbol{\alpha}(t_0) + \dot{\mathbf{q}}(t_0)\boldsymbol{\omega}(t_0), \\ \ddot{\mathbf{q}}(t_f) &= \dot{\mathbf{q}}(t_f)2\boldsymbol{\alpha}(t_f) + \dot{\mathbf{q}}(t_f)\boldsymbol{\omega}(t_f).\end{aligned}\tag{121}$$

Based on the properties of the 5th order Bezier polynomial, the coefficients of the quaternion expression are then calculated by:

$$\tilde{\boldsymbol{\omega}}_1 = \frac{1}{10}\boldsymbol{\omega}(t_0),\tag{122}$$

$$\tilde{\boldsymbol{\omega}}_5 = \frac{1}{10}\boldsymbol{\omega}(t_f),\tag{123}$$

$$\tilde{\boldsymbol{\omega}}_2 = \frac{\tilde{\mathbf{q}}_0^{-1}\ddot{\mathbf{q}}(t_0) + 20\tilde{\boldsymbol{\omega}}_1 - 25\tilde{\boldsymbol{\omega}}_1^2}{20},\tag{124}$$

$$\tilde{\omega}_4 = \frac{\tilde{\mathbf{q}}_4^{-1} \left(\ddot{\mathbf{q}}(t_f) + 20\tilde{\mathbf{q}}_5\tilde{\omega}_1 - 25\mathbf{q}_5\tilde{\omega}_1^2 \right) \tilde{\mathbf{q}}_5^{-1} \tilde{\mathbf{q}}_4}{-20}. \quad (125)$$

Here, $\ddot{\mathbf{q}}$ is computed using the second equation in Equation (119) and the complementary $\tilde{\mathbf{q}}_i$ parameters defined as:

$$\begin{aligned} \tilde{\mathbf{q}}_0 &= \mathbf{q}(t_0) = \mathbf{q}(\tau)|_{\tau=0}, \\ \tilde{\mathbf{q}}_5 &= \mathbf{q}(t_f) = \mathbf{q}(\tau)|_{\tau=1}, \\ \tilde{\mathbf{q}}_1 &= \tilde{\mathbf{q}}_0 \exp(\tilde{\omega}_1), \\ \tilde{\mathbf{q}}_4 &= \tilde{\mathbf{q}}_5 \exp(\tilde{\omega}_5)^{-1}. \end{aligned} \quad (126)$$

and

$$\begin{aligned} \tilde{\mathbf{q}}_2 &= \mathbf{q}_1 \exp(\tilde{\omega}_2), \\ \tilde{\mathbf{q}}_3 &= \mathbf{q}_5 \left(\exp(\tilde{\omega}_4) \exp(\tilde{\omega}_5) \right)^{-1}. \end{aligned} \quad (127)$$

Finally,

$$\tilde{\omega}_3 = \ln(\tilde{\mathbf{q}}_2^{-1} \tilde{\mathbf{q}}_3). \quad (128)$$

The resulting benefit of this laborious formulation is that the attitude trajectory history of a 4x1 \mathbf{q} vector, that satisfies the constraints of a unit quaternion, can be specified by a reduced set of parameters. These parameters are the initial and final conditions on the quaternion itself, as well as values of angular velocity and angular acceleration at those endpoints.

5. Increasing the Polynomial Order

More flexibility in the trajectory can be obtained by increasing the order of the Bezier polynomial used in the basis function. For the case of a 7th order polynomial, the same structure as Equation (104) is employed but now with $n=7$. The resulting 7th order polynomials are shown in Figure 75.

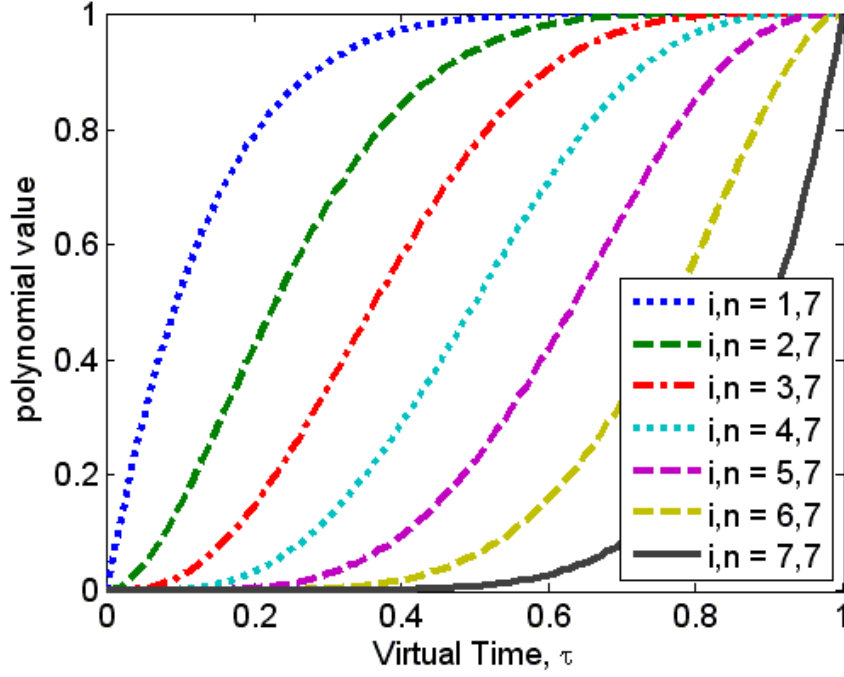


Figure 75. Value of 7th order Bezier Polynomials with respect to the virtual arc.

This leads to the introduction of $\tilde{\mathbf{q}}_6, \tilde{\mathbf{q}}_7, \tilde{\boldsymbol{\omega}}_6$ and $\tilde{\boldsymbol{\omega}}_7$, which are defined to be consistent with previous definitions from Equations (104) and (112). If the values of orientation and angular velocity at the endpoints are set, endpoint conditions of angular jerk as well as angular acceleration can be used as varied parameters. Setting a specified (low) value for the initial and final jerk can be critical for slewing maneuvers of flexible spacecraft, in order to avoid excitation of structural modes. To do this, the third derivative of the output vector is calculated as:

$$\ddot{\mathbf{q}} = \frac{d^3 \mathbf{q}}{d\tau^3} \lambda^3 + \frac{d^2 \mathbf{q}}{d\tau^2} 2\lambda \frac{d\lambda}{d\tau} \lambda + \frac{d^2 \mathbf{q}}{d\tau^2} \frac{d\lambda}{d\tau} \lambda^2 + \frac{d\mathbf{q}}{d\tau} \frac{d^2 \lambda}{d\tau^2} \lambda^2 + \frac{d\mathbf{q}}{d\tau} \left(\frac{d\lambda}{d\tau} \right)^2 \lambda. \quad (129)$$

The new expressions for the virtual derivatives are also recalculated, which are analogous to Equations (114) and (115) but with the addition of a third derivative to accommodate the change between 5th and 7th order polynomial:

$$\left. \frac{d^3 \mathbf{q}}{d\tau^3} \right|_{\tau=0} = 343\tilde{\mathbf{q}}_0 \tilde{\omega}_1^3 - 882\tilde{\mathbf{q}}_0 \tilde{\omega}_1^2 - 420\tilde{\mathbf{q}}_0 \tilde{\omega}_2 + 210\tilde{\mathbf{q}}_0 \tilde{\omega}_1 + 210\tilde{\mathbf{q}}_0 \tilde{\omega}_3 + 882\tilde{\mathbf{q}}_0 \tilde{\omega}_1 \tilde{\omega}_2, \quad (130)$$

$$\begin{aligned} \left. \frac{d^3 \mathbf{q}}{d\tau^3} \right|_{\tau=1} &= 343\tilde{\mathbf{q}}_7 \tilde{\omega}_7^3 - 882\tilde{\mathbf{q}}_6 \tilde{\omega}_6 \tilde{\omega}_7 \tilde{\mathbf{q}}_6^{-1} \tilde{\mathbf{q}}_7 + 210\tilde{\omega}_5 \tilde{\mathbf{q}}_7 + 210\tilde{\mathbf{q}}_7 \tilde{\omega}_7 + 882\tilde{\mathbf{q}}_7 \tilde{\omega}_7^2 \\ &\quad + 882\tilde{\mathbf{q}}_0 \tilde{\omega}_1 \tilde{\omega}_2 - 420\tilde{\omega}_6 \tilde{\mathbf{q}}_7; \end{aligned}$$

$$\begin{aligned} \left. \frac{d^2 \mathbf{q}}{d\tau^2} \right|_{\tau=0} &= 49\tilde{\mathbf{q}}_0 \tilde{\omega}_1^2 + 42\tilde{\mathbf{q}}_0 \tilde{\omega}_2 - 42\tilde{\mathbf{q}}_0 \tilde{\omega}_1, \\ \left. \frac{d^2 \mathbf{q}}{d\tau^2} \right|_{\tau=1} &= 42\tilde{\mathbf{q}}_7 \tilde{\omega}_7 + 49\tilde{\mathbf{q}}_7 \tilde{\omega}_7^2 - 42\tilde{\omega}_6 \tilde{\mathbf{q}}_7; \end{aligned} \quad (131)$$

$$\begin{aligned} \left. \frac{d\mathbf{q}}{d\tau} \right|_{\tau=0} &= 7\tilde{\mathbf{q}}_0 \tilde{\omega}_1, \\ \left. \frac{d\mathbf{q}}{d\tau} \right|_{\tau=1} &= 7\tilde{\mathbf{q}}_7 \tilde{\omega}_7. \end{aligned} \quad (132)$$

New values for the constants that fix the initial conditions of the quaternion trajectory can be calculated similar to the 5th order polynomial, except an extra step needs to be taken to accommodate the third derivative of \mathbf{q} .

D. SOLVING THE PROBLEM USING IDVD METHOD

This section presents the results of using IDVD method with two different parameterizations to obtain the minimum time solutions of the problem posed in Chapter III.A. The Matlab function *fmincon* was used to optimize the trajectory while varying either the angular acceleration (for the 5th order polynomial) or the angular acceleration and jerk (for the 7th order polynomial) at the endpoints of the trajectory. Together with five varied parameters for $\lambda(t)$, Equation (117), it yields 11 varied parameters for the 5th order polynomial approximation and 17—for the 7th order polynomial approximation. The constraints are that the resulting control must obey Equation (102) and $\lambda(t)$ cannot

be ≤ 0 at any instant. Initial guesses for the angular acceleration and jerk are equal to zero and the initial guess of 10^{-4} is used for all parameters contained in $\lambda(t)$.

1. IDVD Solutions Varying 2nd Derivative Results

Figures 76–79 present the results obtained when applying the IDVD with a quaternion based on a 5th order Bezier polynomial (compare it with the GPOPS solution presented in Figures 66–68), allowing variation of the 2nd derivative at the endpoints. The solution was evaluated using 100 nodes (although, as opposed to GPOPS because the solution is analytic, it would be no difference running it for a larger or smaller number of nodes), resulted in slightly larger t_f but took significantly less time to compute at only 10.0 seconds.

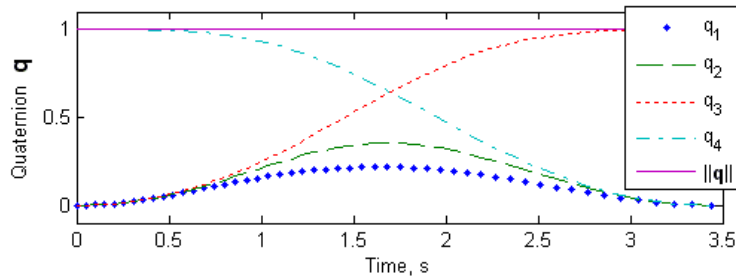


Figure 76. Case 1 (IDVD 5th order): time history of the quaternion.

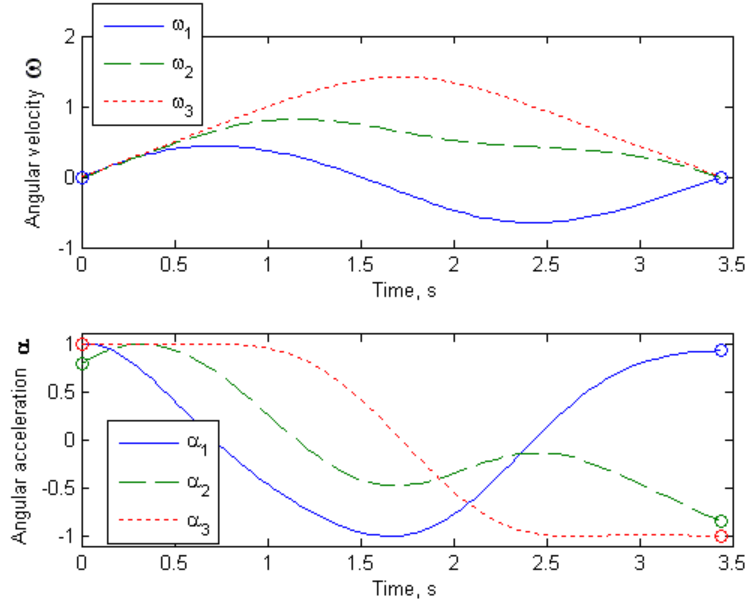


Figure 77. Case 1 (IDVD 5th order): time histories of the angular velocity and accelerations.

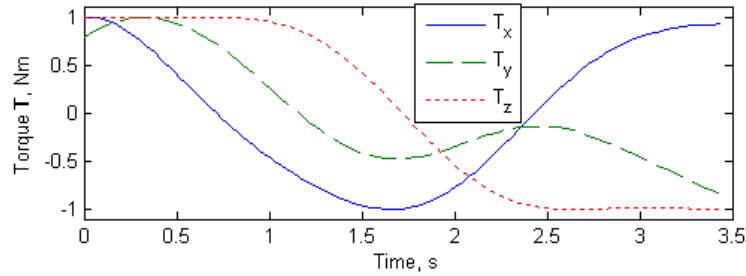


Figure 78. Case 1 (IDVD 5th order): time history of controlling torques.

The major difference compared to the GPOPS solution is that the controls do not have a bang-bang nature anymore. Again, it was done intentionally by the choice of the quaternion parameterization. It occurs that when implemented in the real time controller, these controls will be easier to track. Also, having different controlling torques at the endpoints means having different angular accelerations. While for the GPOPS solution, the terminal angular accelerations are at the mercy of the optimization routine using IDVD allows matching them with the current accelerations, which also makes the control algorithm more robust. Figure 78 shows the speed factor, the key element in matching the virtual and time domains.

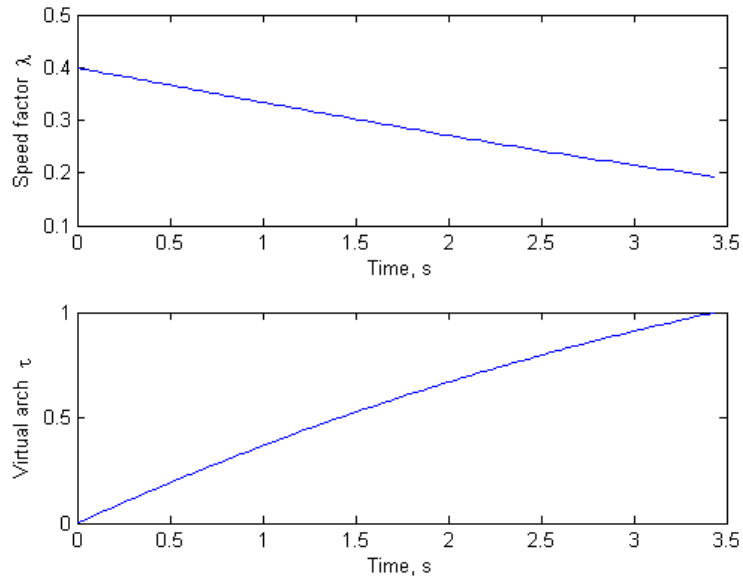


Figure 79. Case 1 (IDVD 5th order): mapping the virtual and time.

Figure 80 shows an outline of the slew in inertial space, clearly showing that it is not an eigenaxis maneuver.

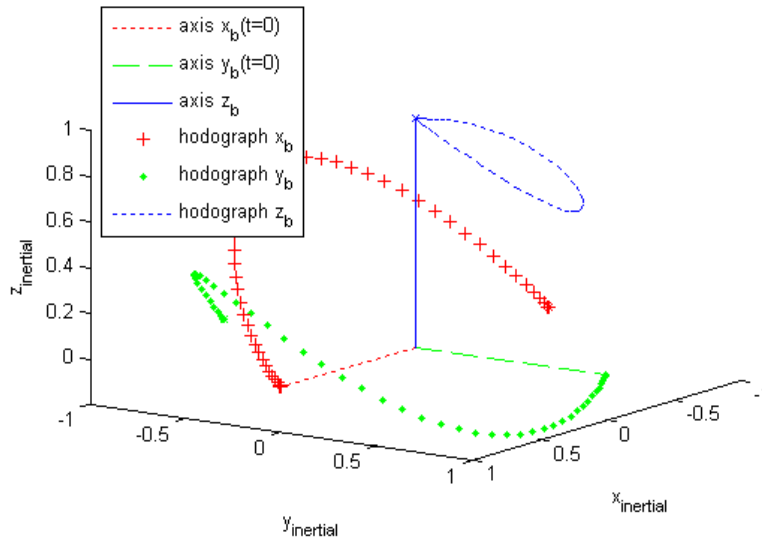


Figure 80. Case 1 (IDVD 5th order): principal axis outline of 180° slew.

The resulting solution for the same scenario using 25 nodes is shown in Figures 81–84.

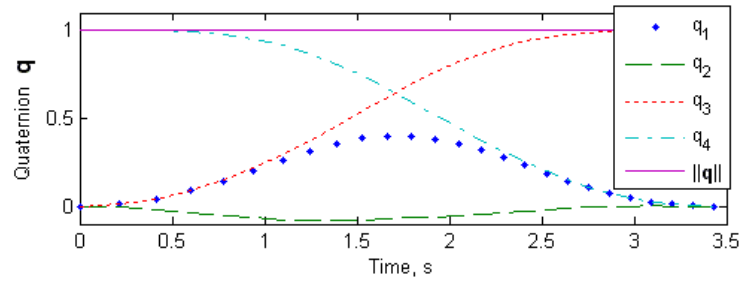


Figure 81. Case 1 (IDVD 5th order): time history of the quaternion using 25 nodes.

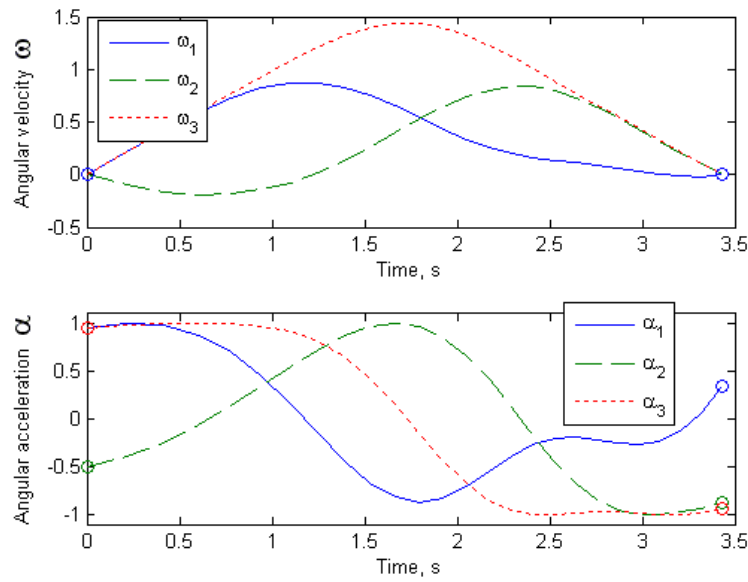


Figure 82. Case 1 (IDVD 5th order): time histories of the angular velocity and acceleration using 25 nodes.

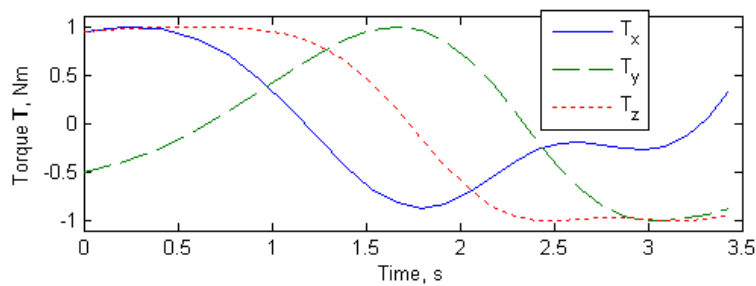


Figure 83. Case 1 (IDVD 5th order): time history of controlling torques using 25 nodes.

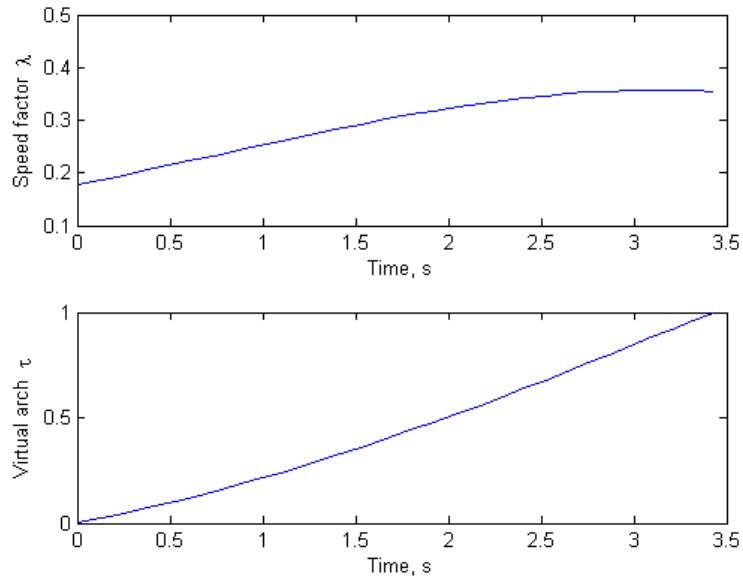


Figure 84. Case 1 (IDVD 5th order): mapping the virtual and time domains using 25 nodes.

The rate profiles for the solution of 25, 50, and 100 nodes are shown in Figures 85 and 86. Each case results in a smooth control solution regardless of the number of nodes chosen. This is due to the construction of the quaternion history as well as the controls.

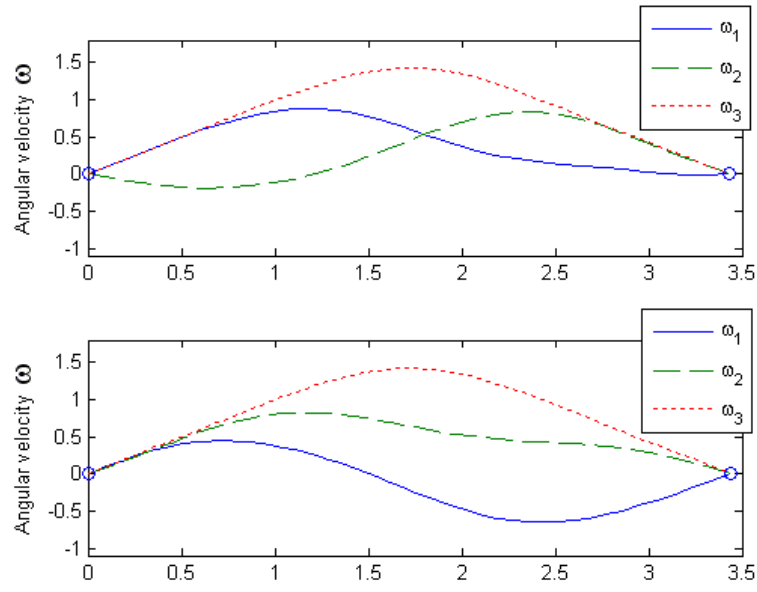


Figure 85. Case 1 (IDVD 5th order): comparison of angular velocity time histories, obtained for 25 nodes (top) and 50 nodes (bottom).

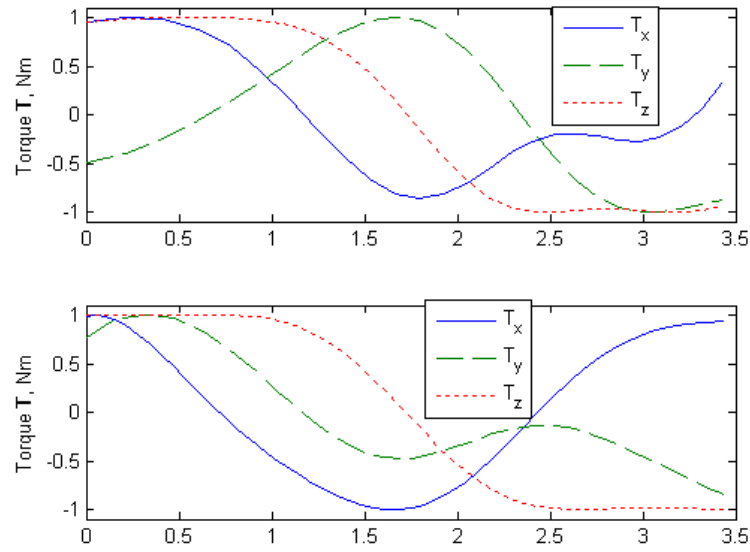


Figure 86. Case 1 (IDVD 5th order): comparison of torque time histories, obtained for 25 nodes (top) and 50 nodes (middle).

As in the case of GPOPS solution for Case 2, the nonsymmetrical inertia matrix causes a certain changes as compared to the symmetric matrix solution. The 100-node IDVD solution in this case results in 4.767s maneuver and requires about a minute to compute (see Figures 87–90).

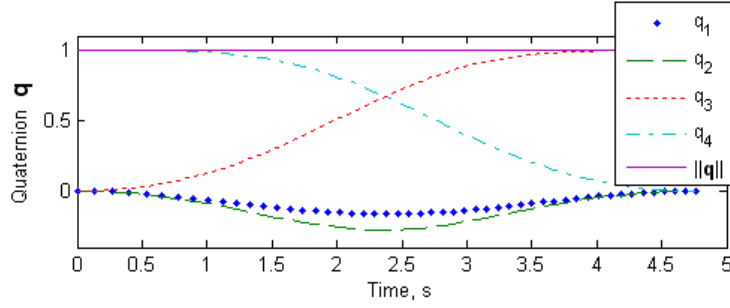


Figure 87. Case 2 (IDVD 5th order): Time history of the quaternion.

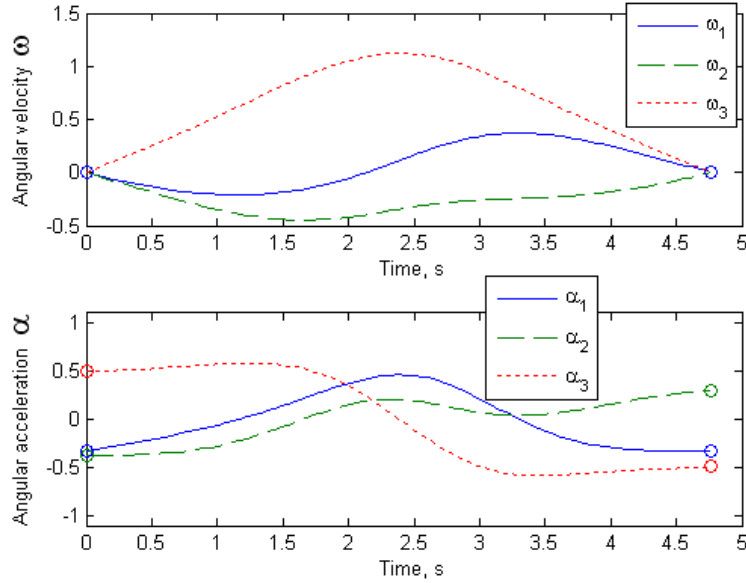


Figure 88. Case 2 (IDVD 5th order): time histories of the angular velocity and acceleration.

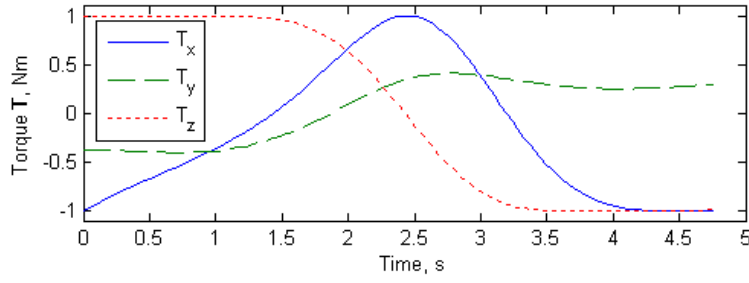


Figure 89. Case 2 (IDVD 5th order): time history of controlling torques.

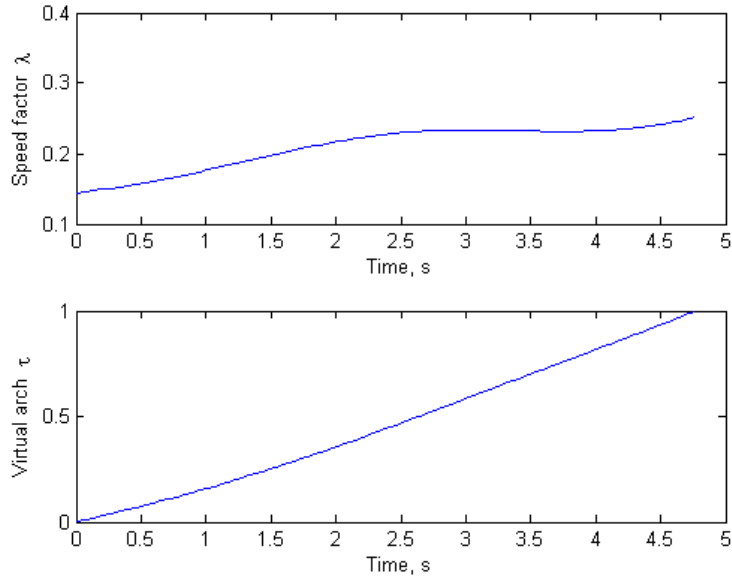


Figure 90. Case 2 (IDVD 5th order): mapping the virtual and time domains.

2. IDVD Solutions Varying 2nd–3rd Derivative Results

For the sake of comparison, Figures 91–94 present the solution of the same problem using a quaternion based on a 7th order Bezier polynomial, allowing variation of both the 3rd and 4th derivative at the endpoints. Although this slightly improves the PI, it drastically increases the computational time. The following section addresses this issue in more detail.

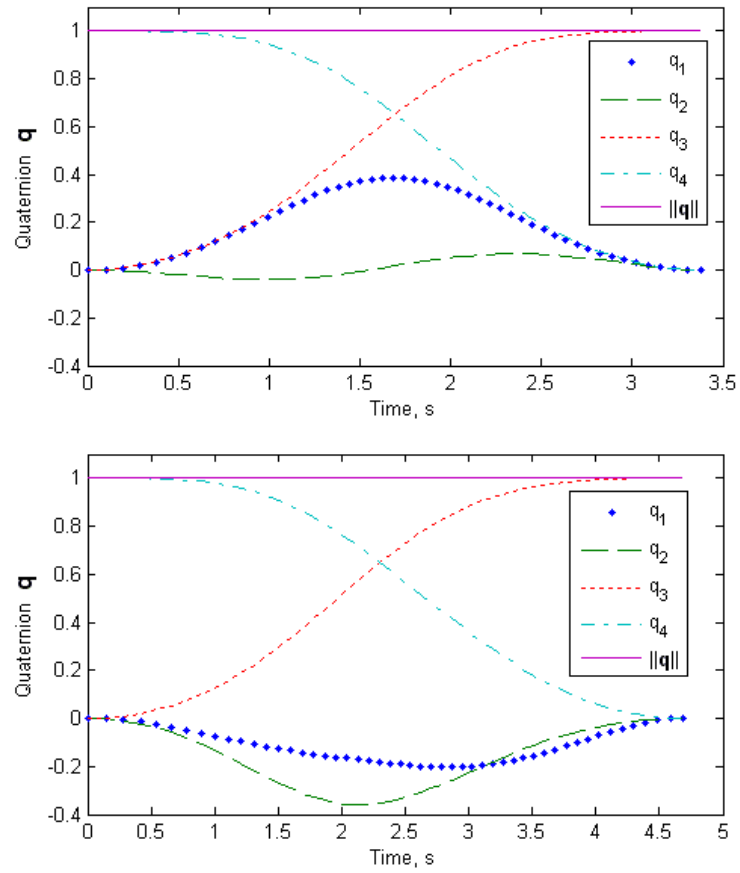


Figure 91. Case 1 and 2 (IDVD 7th order): time history of the quaternions Case 1 (top) and Case 2 (bottom).

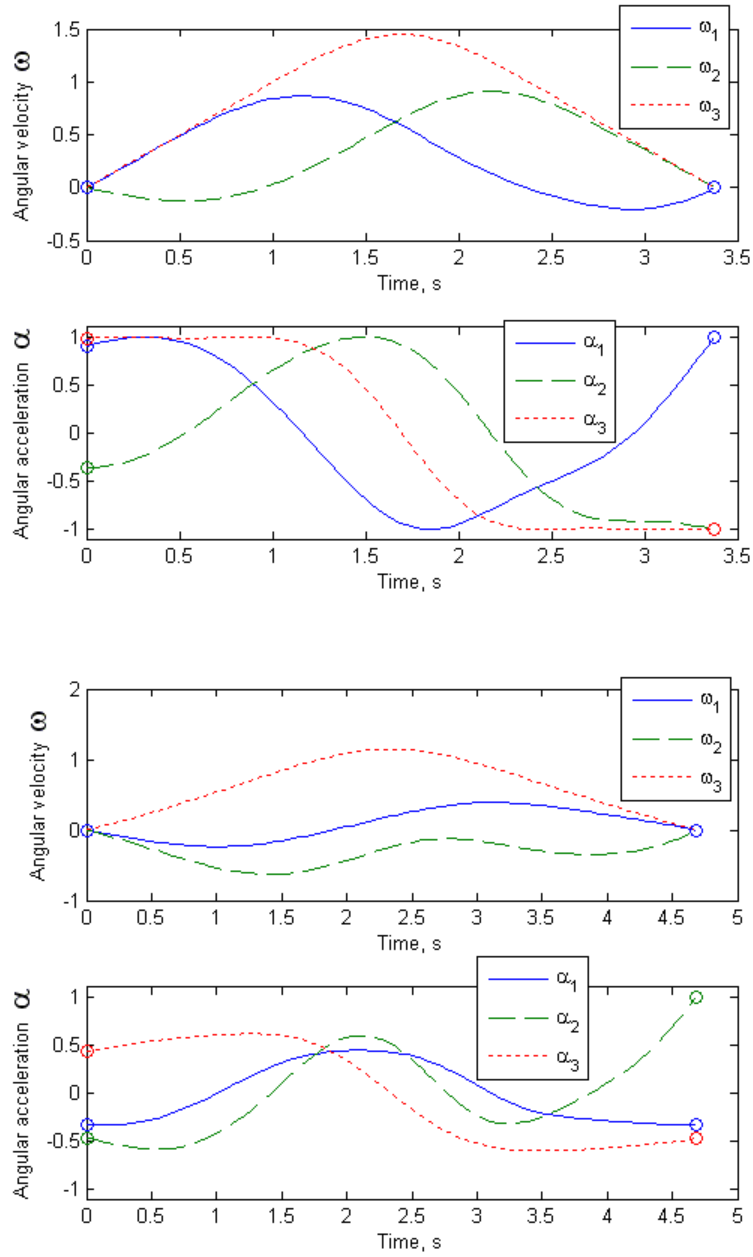


Figure 92. Case 1 and 2 (IDVD 7th order): time histories of the angular velocity and acceleration for Case 1 (top) and Case 2 (bottom).

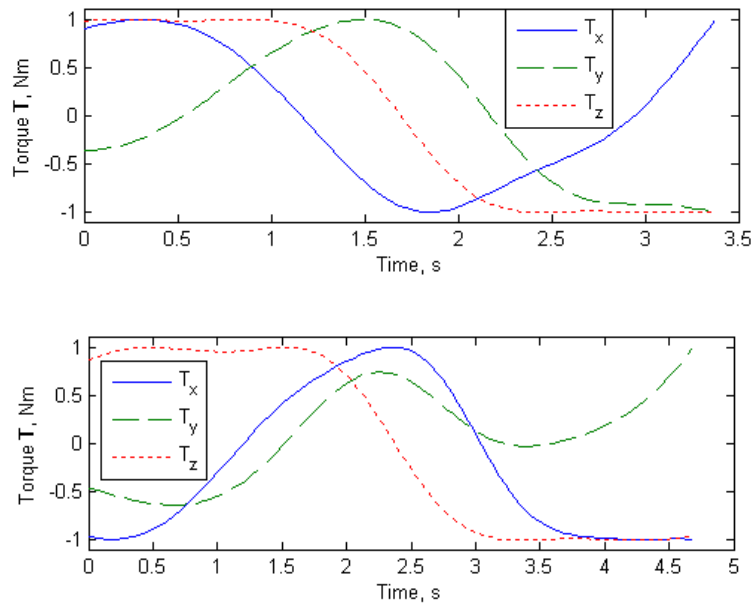


Figure 93. Case 1 and 2 (IDVD 7th order): time history of controlling torques for Case 1 (top) and Case 2 (bottom).

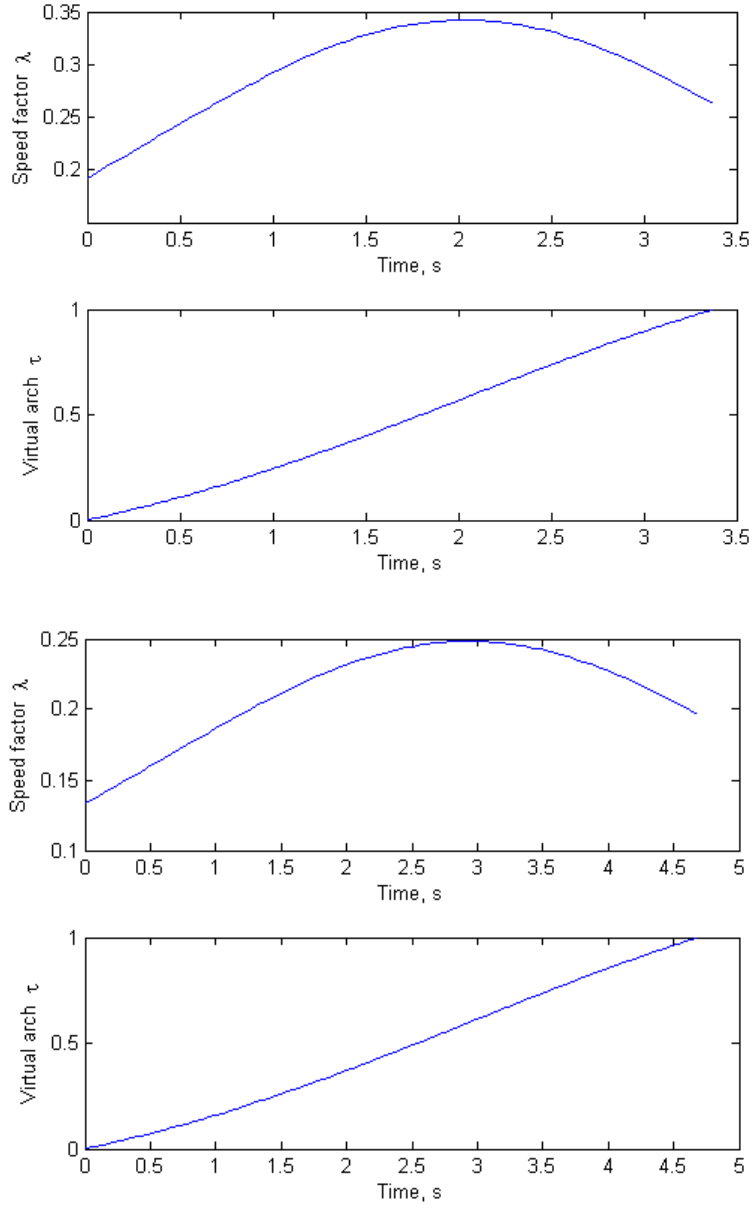


Figure 94. Case 1 and 2 (IDVD 7th order): mapping the virtual and time domains for Case 1 (top) and Case 2 (bottom) solutions.

E. RESULTS AND COMPARISONS

This section presents a comparison of the results obtained using the IDVD method with those of the GPOPS method. It disregards the fact that the results obtained with GPOPS for low number of nodes are infeasible but rather concentrates on the computational advantages the IDVD approach has for any number of intermediate points

(nodes in the case of GPOPS). To start with, Tables 17 and 18 summarize the 180° rest-to-rest slew maneuver solutions for symmetric and asymmetric inertia matrix obtained using GPOPS and IDVD, as discussed in Chapters IV.C and IV.D.

In these tables, all results are compared against the readily available eigenaxis maneuver solution. First, it is shown that the true optimal solution, obtained offline (Bilimoria and Wie 1993), which is not an eigenaxis rotation, provides about 8.5% and 11% improvement of the PI, time of maneuver, for Case 1 and Case 2, respectively.

As seen from Tables 17 and 18, the GPOPS solution converging to one of the equally optimal solutions (if at all), assures about the same gain in the PI as the truly optimal one. But, again, it takes significant computational time. Specifically, the 100-node solution that does converge and assures a smooth controls history takes about an hour to converge.

Trajectory Generation Method	Nodes	Computational Time (sec)	cost (t_f)	% improvement
Eigenaxis	N/A	N/A	3.5449	~0%
Optimal (Bilimoria/Wie)	N/A	N/A	3.2431	8.51%
GPOPS	25	15.5	3.2573	8.11%
GPOPS	50	200.5	3.2859	7.31%
GPOPS	100	5962.8	3.2430	8.52%
IDVD 5th order	25	3.7	3.4289	3.27%
IDVD 5th order	50	4.8	3.4373	3.04%
IDVD 5th order	100	10.0	3.4382	3.01%
IDVD 7th order	25	44.0	3.3756	4.78%
IDVD 7th order	50	69.8	3.3769	4.74%
IDVD 7th order	100	121.2	3.3776	4.72%

Table 17. The 180° rest-to-rest slew maneuver about the z-axis, symmetric inertia (Case 1).

Trajectory Generation Method	Nodes	Computational Time (sec)	cost (t_f)	% improvement
Eigenaxis	N/A	~0	5.0133	~0%
GPOPS	25	22.2	4.4609	11.02%
GPOPS	50	520.1	4.4499	11.24%
GPOPS	100	1893.9	4.4528	11.18%
IDVD 5th order	25	4.8	4.7857	4.54%
IDVD 5th order	50	7.6	4.7861	4.53%
IDVD 5th order	100	9.5	4.7864	4.53%
IDVD 7th order	25	24.3	4.6768	6.71%
IDVD 7th order	50	41.2	4.6846	6.56%
IDVD 7th order	100	80.4	4.6869	6.51%

Table 18. The 180° rest-to-rest slew maneuver about the z-axis, asymmetric inertia (Case 2).

As discussed in the previous section, the IDVD solution has a much more robust performance, allowing computing the same type of maneuvers just in a few seconds as opposed to hours. If an executable optimization library is implemented, the IDVD method produces solutions in fractions of a second (Yakimenko and Slegers 2009). Of course, some of the optimality (PI value) is sacrificed. On the positive side, the solution is always feasible and smooth for any number of computational points, and can be brought closer to the GPOPS solution (in terms of the value of a PI) by increasing the number of varied parameters (the order of the quaternion approximation polynomial). Furthermore, since the resulting IDVD solution is analytic in nature, increasing the node points, after a solution is obtained, is a trivial evaluation of an analytic expression.

As discussed in Chapter IV.B, this solution features a bang-bang control, i.e., does not account for controllers' dynamics, and therefore can still not be used onboard as is. On the other hand, the always-feasible and ready-to-go IDVD solution (employing as low as say 25 computational points) can be produced much faster, but surrenders up to $\frac{2}{3}$ of its gain as compared to that of the GPOPS solution (about $\frac{1}{2}$ for the 7th order approximation).

Tables 19 and 20 present similar data for the 90° rest-to-rest slew maneuver. While GPOPS provides about 3% gain compared to a simple eigenaxis slew solution, the IDVD method has almost no advantage or may be even worse if using a 5th order quaternion approximation. This is because, at some point, the gains made by having the

ability to calculate a bang-bang solution (with discontinuous controls) outweighs the benefits of having the ability to exploit the gains made by the effect of slewing off the eigenaxis. All major conclusions, however, remain the same.

Trajectory Generation Method	Nodes	Computational Time (sec)	cost (t_f)	% improvement
Eigenaxis	N/A	N/A	2.5066	~0%
GPOPS	25	20.7	2.4336	2.91%
GPOPS	50	120.0	2.4332	2.93%
GPOPS	100	236.2*	2.4296	3.07%
IDVD 5th order	25	5.8	2.5654	-2.34%
IDVD 5th order	50	7.2	2.5666	-2.39%
IDVD 5th order	100	9.4	2.5671	-2.41%
IDVD 7th order	25	39.7	2.5043	0.09%
IDVD 7th order	50	58.9	2.5058	0.03%
IDVD 7th order	100	77.3	2.5058	0.03%

Table 19. The 90° rest-to-rest slew maneuver about the z-axis, symmetric inertia (Case 1).

Tables 21 and 22 compare GPOPS and IDVD solutions for such case, when other sets of nonzero boundary conditions were explored as well, and proved to maintain the same trends. In this table, all results are compared against a valid 100-node GPOPS solution. As seen, the GPOPS solutions with a lesser number of nodes produce somewhat infeasible solutions, meaning that they cannot be implemented in the control scheme explicitly. The IDVD solutions may yield to GPOPS as much as about 4% with respect to the PI, but again are produced much faster. Furthermore, as shown in Figures 95 and 96, the 90° and 180 ° rest-to-rest slew maneuvers with zero boundary rates feature multiple equally optimal solutions, so that both GPOPS and IDVD solutions converge to different solutions when changing the number of nodes (GPOPS) / computational points (IDVD). In contrast, for the case of nonzero boundary rates, they all converge to the same solution, as shown in Figure 97.

Trajectory Generation Method	Nodes	Computational Time (sec)	cost (t_f)	% improvement
Eigenaxis	N/A	~0	3.5449	~0%
GPOPS	25	17.4	3.4450	2.82%
GPOPS	50	168.0	3.4408	2.94%
GPOPS	100	1432.1	3.4430	2.87%
IDVD 5th order	25	5.9	3.6277	-2.33%
IDVD 5th order	50	9.8	3.6307	-2.42%
IDVD 5th order	100	17.3	3.6316	-2.44%
IDVD 7th order	25	41.5	3.5373	0.21%
IDVD 7th order	50	80.4	3.5389	0.17%
IDVD 7th order	100	158.8	3.5394	0.16%

Table 20. The 90° rest-to-rest slew maneuver about the z-axis, asymmetric inertia (Case 2).

It should be noted that in practice, the direct methods would likely be used in situations where the end-conditions (angular rates, accelerations) of the slew are specified and not equal to zero (to meet mission requirements of matching attitude rates of a tumbling vehicle, for example). For this case, no simple eigenaxis slew solution exists and therefore any solution produced on-line would be good. The next chapter of this dissertation exploits the IDVD and coupling with translational maneuvers.

Trajectory Generation Method	Nodes	Computational Time (sec)	cost (t_f)	% within GPOPS 100 nodes
GPOPS	25	48.3	2.4028	0.07%
GPOPS	50	333.9	2.4016	0.02%
GPOPS	100	1182.6	2.4011	0.00%
IDVD 5th order	25	5.9	2.5437	5.94%
IDVD 5th order	50	5.6	2.5450	5.99%
IDVD 5th order	100	6.1	2.5451	6.00%
IDVD 7th order	25	27.9	2.4885	3.64%
IDVD 7th order	50	41.5	2.4895	3.68%
IDVD 7th order	100	90.6	2.4896	3.69%

Table 21. The 90° maneuver for symmetric inertia (Case 1) and nonzero boundary rates,

$$\boldsymbol{\omega}_0 = -\boldsymbol{\omega}_f = [0.1 \ 0.1 \ 0.1]^T.$$

Trajectory Generation Method	Nodes	Computational Time (sec)	cost (t_f)	% within GPOPS 100 nodes
GPOPS	25	11.7	2.9054	0.03%
GPOPS	50	111.3	2.9047	0.01%
GPOPS*	100	164.3	2.9045	0.00%
IDVD 5th order	25	6.3	3.0060	3.49%
IDVD 5th order	50	5.5	3.0064	3.51%
IDVD 5th order	100	8.3	3.0066	3.52%
IDVD 7th order	25	31.4	2.9559	1.77%
IDVD 7th order	50	60.1	2.9568	1.80%
IDVD 7th order	100	106.5	2.9571	1.81%

Table 22. The 90° maneuver for symmetric inertia (Case 1) and nonzero boundary rates,

$$\boldsymbol{\omega}_0 = -\boldsymbol{\omega}_f = -\begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T.$$

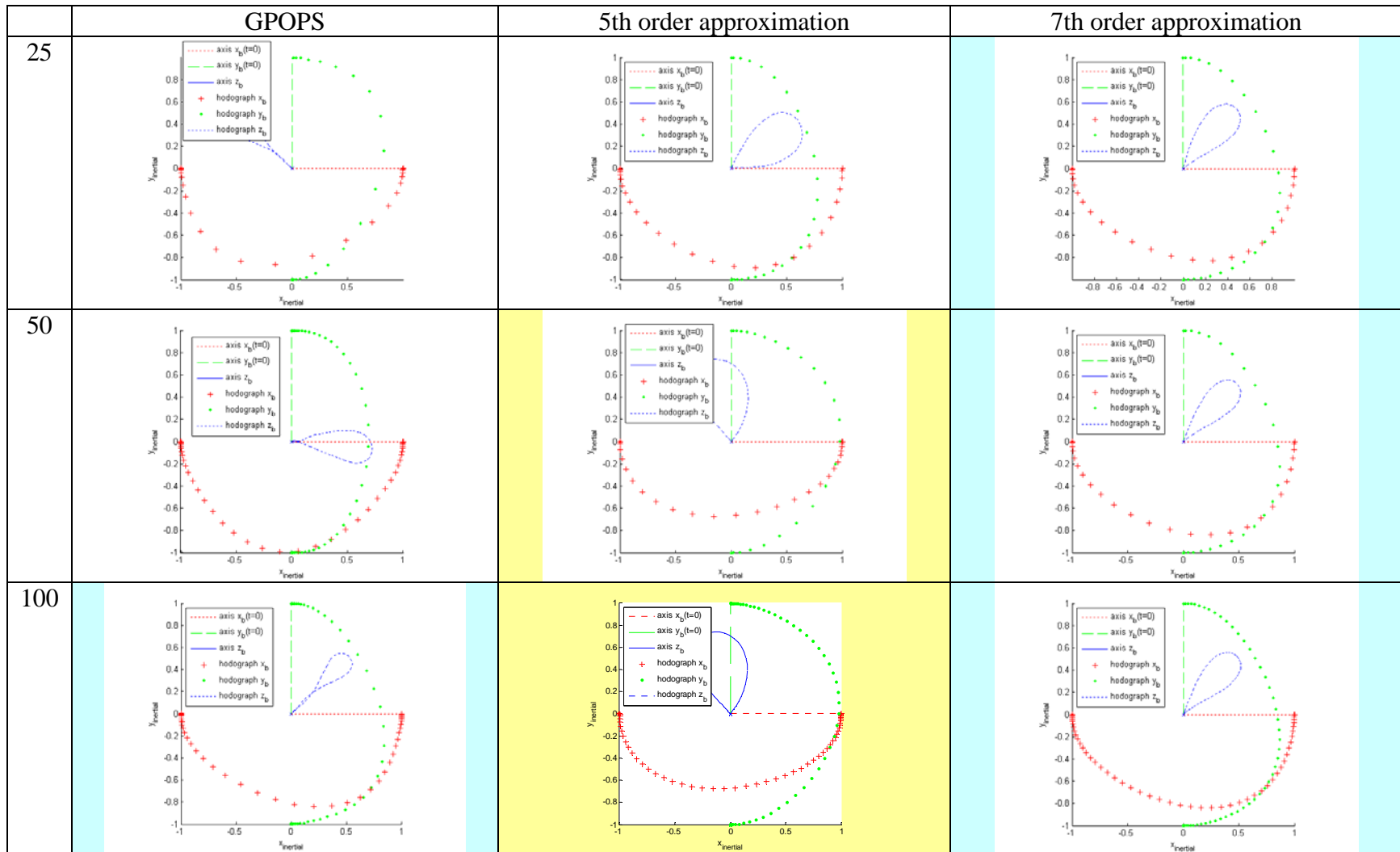


Figure 95.

Case 1: 180° rest-to-rest slew profile as projected onto the x-y inertial plane.

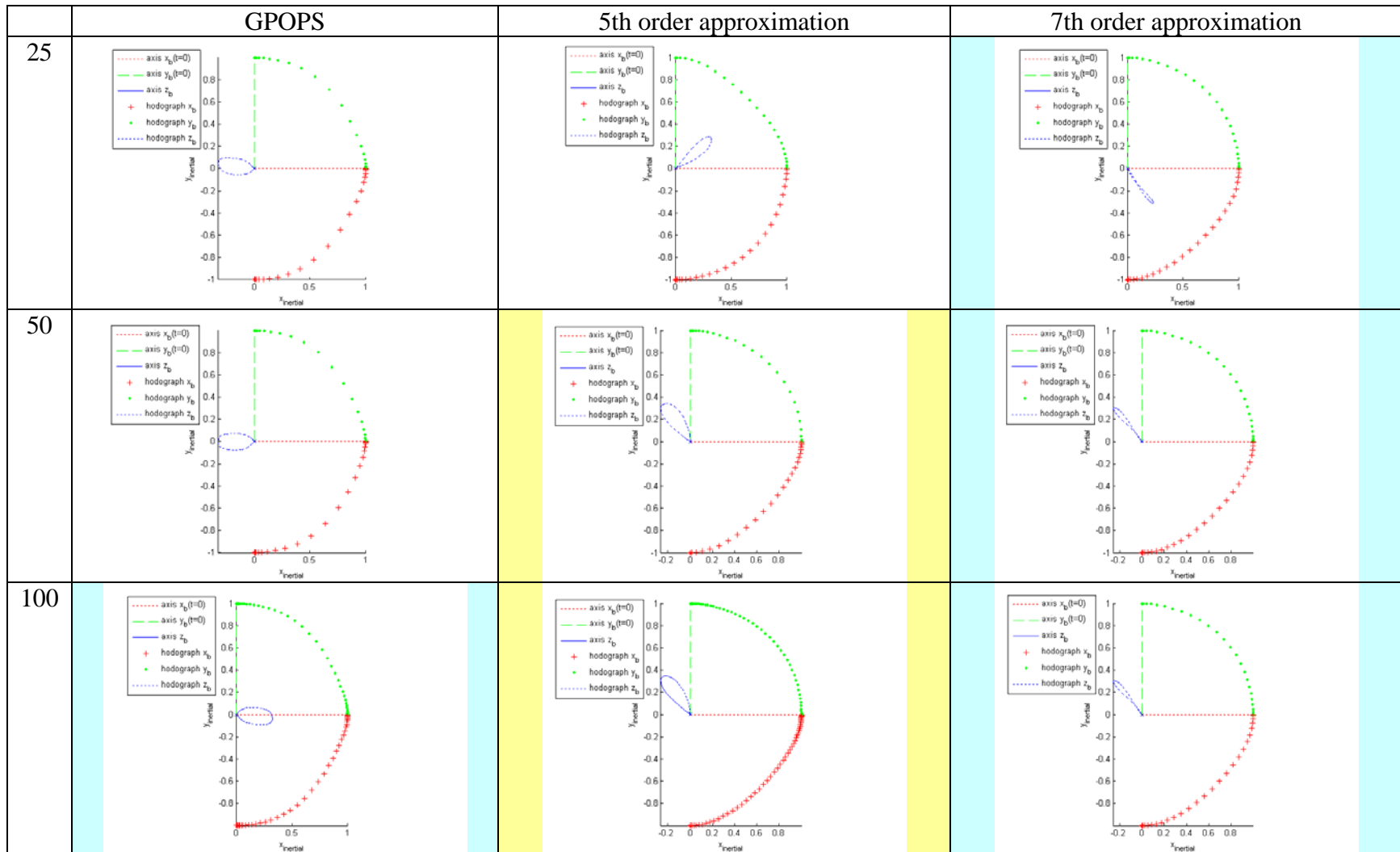


Figure 96. Case 1: 90° rest-to-rest slew profile as projected onto the x-y inertial plane.

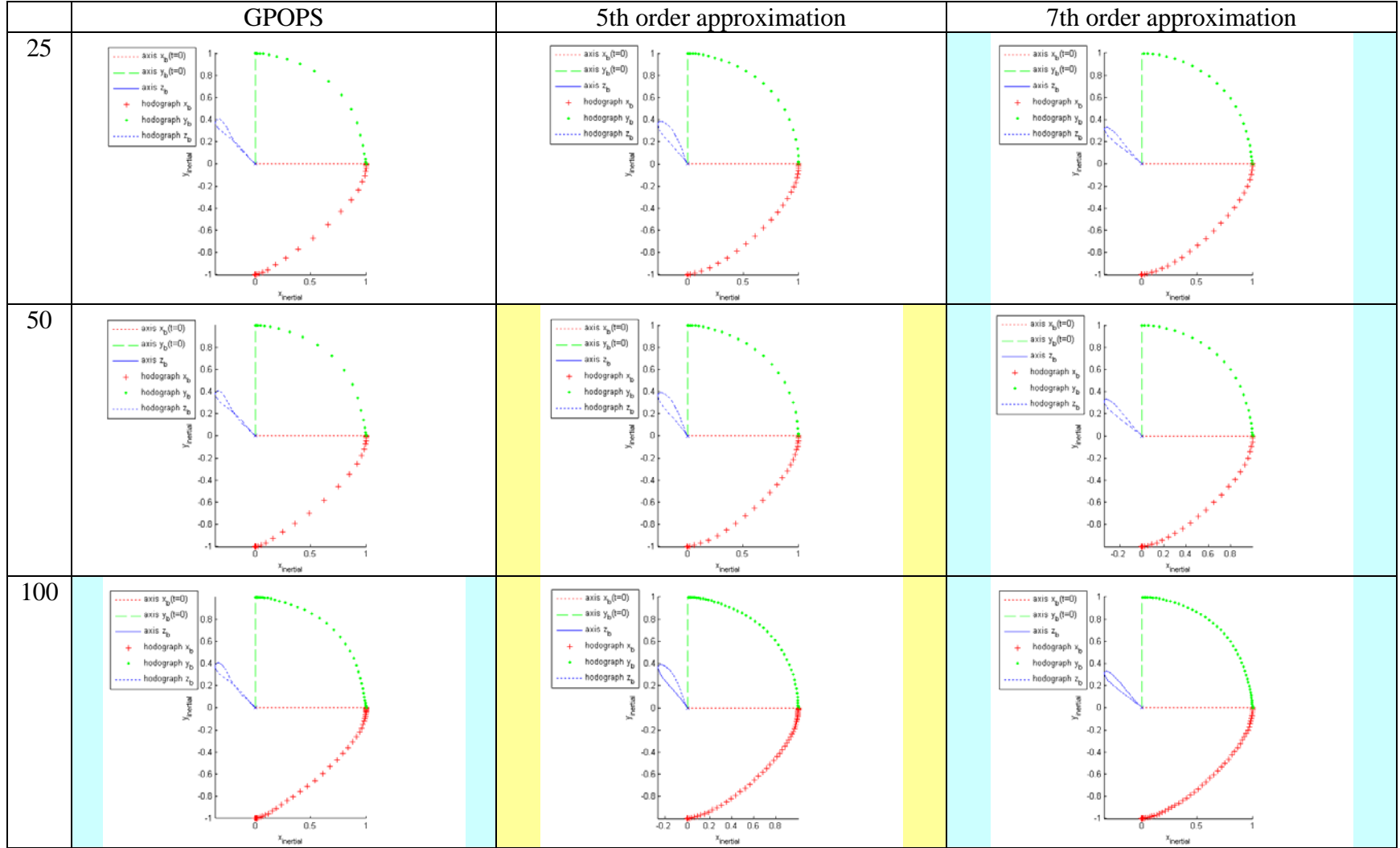


Figure 97. Case 1: 90° slew profile, $\boldsymbol{\omega}_0 = -\boldsymbol{\omega}_f = [0.1 \ 0.1 \ 0.1]^T$, as projected onto the x-y inertial plane.

V. RAPID ATTITUDE AND TRANSLATIONAL TRAJECTORY GENERATION

Inverse Dynamics in the Virtual Domain exploits the ability to take a polynomial of sufficiently high order and match desired values, and potentially derivative values, at its endpoints to generate a trajectory. This is done by representing each position variable of the trajectory of the close approach maneuver in the orbital frame by a summation of polynomials and trigonometric basis functions. While some endpoint conditions are specified based on the problem statement, higher order derivative of the endpoints can function as parameters that may be varied to optimize the trajectory based on a chosen PI. The remainder of this chapter explains the process in detail and applies the IDVD method to the benchmark problem stated in Chapter III.

A. INVERSE DYNAMICS IN THE VIRTUAL DOMAIN FORMULATION

First, a basis function for the translational trajectory is defined by Equation (133).

$$p(\tau) = a_0 + a_1\tau + a_2\tau^2 + a_3\tau^3 + b_0 \sin\left(\frac{\pi}{2}\tau\right) + c_0 \cos\left(\frac{\pi}{2}\tau\right) + b_1 \sin(\pi\tau) + c_1 \cos(\pi\tau) . \quad (133)$$

This leads to 3 separate equations (and separate set of coefficients) for the x , y , and z position of the chaser vehicle each based on the form of the basis function described by Equation (133). It should be reinforced that the trajectory described by Equation (133) with higher order derivatives shown by Equations (134)–(135), are constructed with respect to a virtual argument, $\tau \in [0; \tau_f]$, and not time. This specifies the trajectory in the spatial domain, but lets the speed the trajectory is traversed to be varied as well.

$$p' = \frac{dp}{d\tau} = \begin{pmatrix} a_1 + 2a_2\tau + 3a_3\tau^2 + b_0 \frac{\pi}{2} \cos\left(\frac{\pi}{2}\tau\right) - c_0 \frac{\pi}{2} \sin\left(\frac{\pi}{2}\tau\right) \\ + b_1 \pi \cos(\pi\tau) - c_1 \pi \sin(\pi\tau) \end{pmatrix} \quad (134)$$

$$p'' = \frac{d^2p}{d\tau^2} = \begin{pmatrix} a_2 + 6a_3\tau - b_0 \frac{\pi^2}{4} \sin\left(\frac{\pi}{2}\tau\right) + c_0 \frac{\pi^2}{4} \cos\left(\frac{\pi}{2}\tau\right) \\ - b_1 \pi^2 \sin(\pi\tau) - c_1 \pi^2 \cos(\pi\tau) \end{pmatrix} \quad (135)$$

The plots of the basis functions considered are shown in Figure 98:

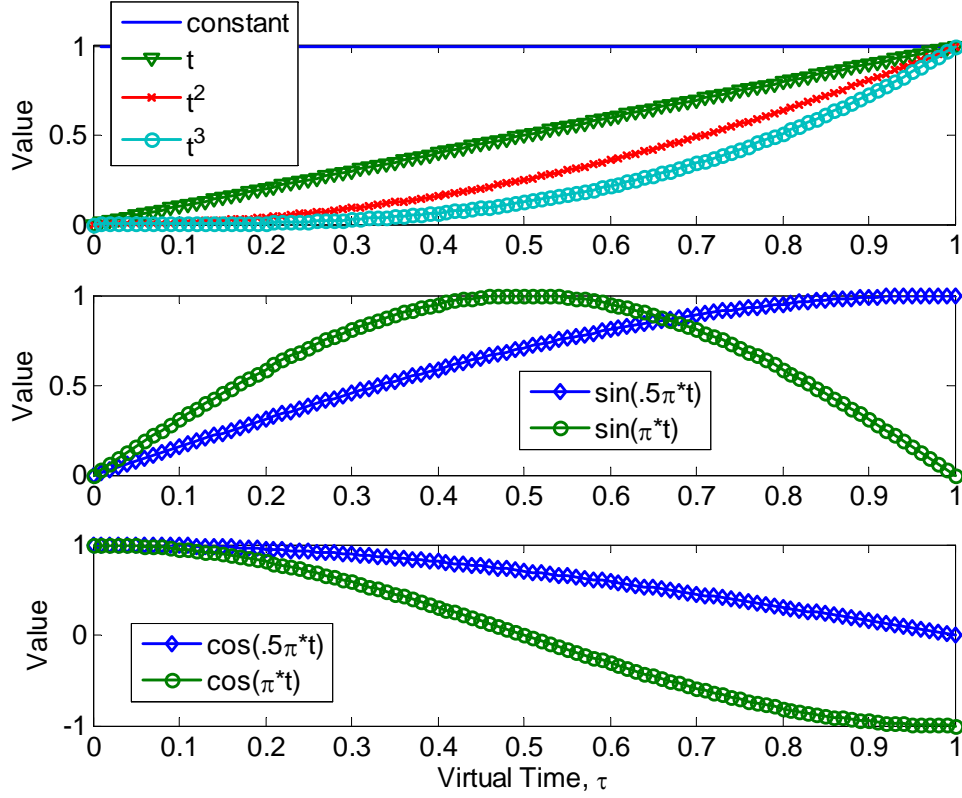


Figure 98. Basis functions considered for translational trajectory generation.

1. Mapping the Virtual Arc and Specifying Endpoint Values

The same structure of the virtual time argument is used as in Chapter IV.B.2, Equation (117). This allows for 5 varied parameters that determine the history of the speed factor, λ , over the duration of the maneuver. The endpoint values of the basis functions are simply set to the beginning and ending values of the maneuver in the spatial domain. The derivative values at the endpoints and their derivatives in the virtual domain are calculated based on the time derivatives at the endpoints and speed factor shown in Equations (136) and (137).

$$\frac{dp_i(t_f)}{d\tau} = \dot{p}_i(t_f) / \lambda(t_f), \quad \frac{dp_i(t_0)}{d\tau} = \dot{p}_i(t_0) / \lambda(t_0) \quad (136)$$

for $i = x, y, z$

and

$$\begin{aligned}\frac{d^2 p_i(t_f)}{d\tau^2} &= (\ddot{p}_i(t_f) - \dot{p}_i(t_f) \frac{d\lambda(t_f)}{dt}) / \lambda(t_f)^2 \\ \frac{d^2 p_i(t_0)}{d\tau^2} &= (\ddot{p}_i(t_0) - \dot{p}_i(t_0) \frac{d\lambda(t_0)}{dt}) / \lambda(t_0)^2\end{aligned}\quad (137)$$

In order to specify the 2nd derivatives at the endpoints, a basis function with a reduction in the number of coefficients is used (Yakimenko 2000; Yakimenko and Slegers 2009):

$$p(\tau) = a_0 + a_1\tau + a_2\tau^2 + c_0 \cos\left(\frac{\pi}{2}\tau\right) + b_1 \sin(\pi\tau) + c_1 \cos(\pi\tau). \quad (138)$$

The resulting coefficients are then be derived according to the algebraic relation in Equation (139).

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & -1 \\ 0 & 1 & 0 & \pi/2 & \pi & 0 \\ 0 & 1 & 2 & 0 & -\pi & 0 \\ 0 & 0 & 2 & 0 & 0 & -\pi^2 \\ 0 & 0 & 2 & -\pi^2/4 & 0 & \pi^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ b_0 \\ b_1 \\ c_1 \end{bmatrix} = \begin{bmatrix} p(t_0) \\ p(t_f) \\ p'(t_0) \\ p'(t_f) \\ p''(t_0) \\ p''(t_f) \end{bmatrix} \quad (139)$$

For higher order basis functions that are capable of specifying jerk at the endpoints, the basis function is specified as Equation (133) and the coefficients are obtained by solving the algebraic relation in Equation (140).

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & \pi/2 & 0 & \pi/2 & 0 \\ 0 & 1 & 2 & 3 & 0 & -\pi/2 & -\pi & 0 \\ 0 & 0 & 2 & 0 & 0 & -\pi^2/4 & 0 & -\pi^2 \\ 0 & 0 & 2 & 6 & -\pi^2/4 & 0 & 0 & \pi^2 \\ 0 & 0 & 0 & 6 & \pi^3/8 & 0 & -\pi^3 & 0 \\ 0 & 0 & 0 & 6 & 0 & \pi^3/8 & \pi^3 & 0 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ b_0 \\ c_0 \\ b_1 \\ c_1 \end{bmatrix} = \begin{bmatrix} p(t_0) \\ p(t_f) \\ p'(t_0) \\ p'(t_f) \\ p''(t_0) \\ p''(t_f) \\ p'''(t_0) \\ p'''(t_f) \end{bmatrix} \quad (140)$$

2. Implementation of Quaternion Based Attitude Representation

The attitude implementation of IDVD for the close rendezvous problem described in Chapter III is done similar to Chapter IV.B. The proper Bezier polynomial is chosen such that the desired boundary conditions can be met as well as allowing variation of the next highest derivative. One major difference is that the attitude trajectory is formulated in the orbital frame instead of the inertial. The angular rates with respect to the inertial frame are then computed, based on Equation (141), and used to determine the resulting control torques in the body frame.

$$\begin{aligned}\omega_x &= {}^o\omega_x + 2(q_1q_3 + q_2q_4)\Omega \\ \omega_y &= {}^o\omega_y + 2(q_2q_3 - q_1q_4)\Omega \\ \omega_z &= {}^o\omega_z + 2(q_4^2 - q_1^2 - q_2^2 + q_3^2)\Omega\end{aligned}\tag{141}$$

3. Alternative Euler Angle Formulation for Attitude Representation

Considering an Euler angle representation of a 2-1-3 with respect to the orbital reference frame rotation can be expressed by the angular displacements θ , ϕ , and ψ pertaining to angular displacements about the y, x and then z body axis respectively (Sidi 1997). The angular rates with respect to the orbit frame can then be expressed as:

$$\begin{aligned}{}^o\omega_x &= \dot{\phi}\cos(\psi) + \dot{\theta}\sin(\psi)\cos(\phi) \\ {}^o\omega_y &= -\dot{\phi}\sin(\psi) + \dot{\theta}\cos(\psi)\cos(\phi) \\ {}^o\omega_z &= \dot{\psi} - \dot{\theta}\sin(\phi)\end{aligned}\tag{142}$$

The angular rates with respect to the inertial frame can then be expressed as:

$$\begin{aligned}\omega_x &= \dot{\phi}\cos(\psi) + \dot{\theta}\sin(\psi)\cos(\phi) + 2(q_1q_3 + q_2q_4)\Omega \\ \omega_y &= -\dot{\phi}\sin(\psi) + \dot{\theta}\cos(\psi)\cos(\phi) + 2(q_2q_3 - q_1q_4)\Omega \\ \omega_z &= \dot{\psi} - \dot{\theta}\sin(\phi) + 2(q_4^2 - q_1^2 - q_2^2 + q_3^2)\Omega\end{aligned}\tag{143}$$

The resulting angular accelerations, with respect to the inertial frame expressed in the body frame becomes:

$$\begin{aligned}
\alpha_x &= \ddot{\phi} \cos(\psi) - \dot{\phi} \dot{\psi} \sin(\psi) + \ddot{\theta} \sin(\psi) \cos(\phi) + \dot{\theta} \dot{\psi} \cos(\phi) \cos(\psi) \\
&\quad - \dot{\theta} \dot{\phi} \sin(\phi) \sin(\psi) + 2(q_3 \dot{q}_1 + q_1 \dot{q}_3 + q_4 \dot{q}_2 + q_2 \dot{q}_4) \omega_o \\
\alpha_y &= -\ddot{\phi} \sin(\psi) - \dot{\phi} \dot{\psi} \cos(\psi) + \ddot{\theta} \cos(\psi) \cos(\phi) - \dot{\theta} \dot{\psi} \cos(\phi) \sin(\psi) \\
&\quad - \dot{\theta} \dot{\phi} \sin(\phi) \cos(\psi) + 2(q_2 \dot{q}_3 + q_3 \dot{q}_2 - q_4 \dot{q}_1 - q_1 \dot{q}_4) \omega_o \\
\alpha_z &= \ddot{\psi} - \dot{\theta} \dot{\phi} \cos(\phi) - \ddot{\theta} \sin(\phi) + 2(q_4 \dot{q}_4 - q_1 \dot{q}_1 - q_2 \dot{q}_2 + q_3 \dot{q}_3) \omega_o
\end{aligned} \tag{144}$$

This equates to the required inertial torque expressed in the body frame dictated by:

$$\begin{aligned}
T_x &= I_{11} \alpha_x + (I_{33} - I_{22}) \omega_z \omega_y \\
T_y &= I_{22} \alpha_y + (I_{11} - I_{33}) \omega_x \omega_z \\
T_z &= I_{133} \alpha_z + (I_{22} - I_{11}) \omega_y \omega_x
\end{aligned} \tag{145}$$

This formulation allows complete description of the attitude trajectory in the orbital and inertial frames and the associated rates and inertial torques by specifying initial and final conditions on θ , ψ , and ϕ as well as their higher order derivatives.

B. SOLVING THE IDVD PROBLEM

For the close rendezvous problem with the IDVD formulation, the Sparse NOlinear OPTimizer (SNOPT) (Gill, Murray, and Saunders 1996) was used to solve for the resulting trajectory. For this implementation, the objective function was the PI based on the specific type of cost function. The problem constraints were programmed as:

$$\begin{aligned}
\text{control constraint: } & -\mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max} \\
\text{path constraint: } & (x^2 + y^2 + z^2) - r^2 \geq 0 \\
\text{time constraints: } & \lambda(\tau) > 0 \quad \forall \tau \\
& t_f \leq t_{f \max}
\end{aligned} \tag{146}$$

with the limits on \mathbf{u} and value of r set to those used in Chapter III. It is reinforced that, with the IDVD method, the dynamic constraints (equations of motion) of the problem and desired endpoint conditions are always satisfied exactly at every iteration. If the problem is not stated such that a solution does not exist, judicious (or sometimes common sense) selection of the initial guess on the varied parameters, conservative guesses on the endpoint derivatives and speed factor coefficients, will lead to always having a feasible

solution to implement. This is not the case with pseudospectral methods, since at any given iteration, the kinematic and dynamic constraints in the equations of motion are not guaranteed to be satisfied and may result in an infeasible solution.

C. COMPUTATIONAL RESULTS

Both IDVD methods of matching endpoint translational and angular position and velocities are used to generate solutions. The 1st set of endpoint parameters is considered to be the position and the 2nd set of endpoint parameters is the velocity. The IDVD (3rd) method refers to varying the set of 3rd parameters, accelerations, at the endpoints while the IDVD (3rd–4th) method refers to varying both the third and fourth parameters, accelerations and jerk.

1. Minimum Energy Cost Varying 3rd Parameter Set Results

The results for using the IDVD (3rd) method for a minimum energy cost, Equation (15), are shown next. They are based on the translational basis Equation (139) and the quaternion attitude formulation employing the 5th order Bezier polynomial. The result is suboptimal when compared to the infinite dimensional optimal control problem solution, but is optimal based on the additional constraint of only using the specified polynomial set of basis functions. This reduces the number of varied parameters to 17 (the third derivative of each state plus the coefficients of the speed factor), resulting in a computational time of 12.2818 seconds. The final PI based on the minimum quadratic-control cost is $J = 0.2461$. Figure 99 shows the 3D representation of the trajectory in Figure 100 shows the planar projection of the same trajectory. The control solution is shown in Figure 101, which is smooth over the interval, consistent with the optimal solution found in Chapter III.C.2. Figures 102 and 103 show the time history of the endpoint discrepancies and verify that the endpoint conditions were met at the final time. The solution using this specific formulation and cost is highly attractive due to its inherent low fuel cost and the attribute that the solution does not demand that the actuators be saturated at any given instant, unlike the bang-bang and bang-off-bang nature of minimum time and minimum fuel cost functions respectively. Specifically, this noncontrol-saturated, smooth trajectory is desirable as the spacecraft is in the final stages

of matching docking points. The speed factor, a representative measure of how fast the chaser is moving along the spatial trajectory, is shown in Figure 104.

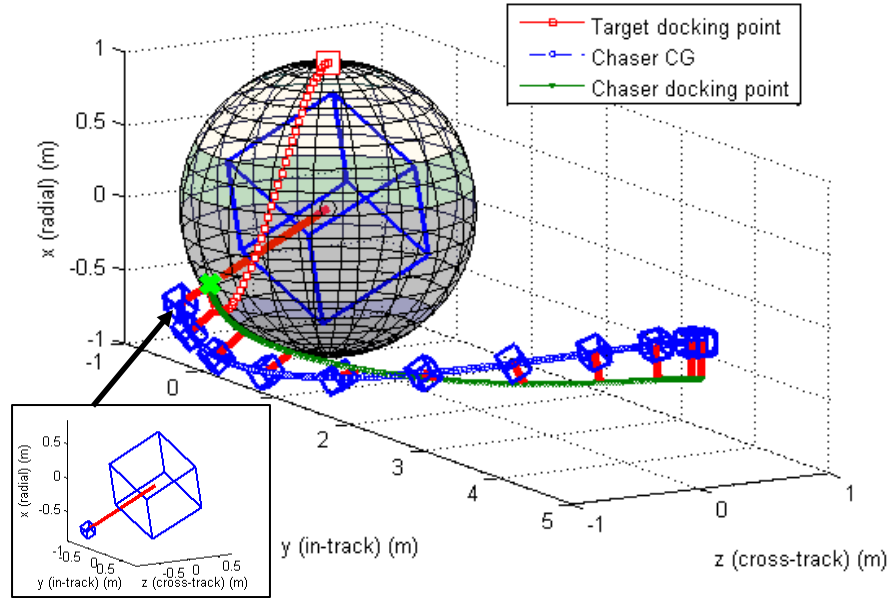


Figure 99. Minimum energy solution (IDVD 3rd): 3D optimal rendezvous trajectory.

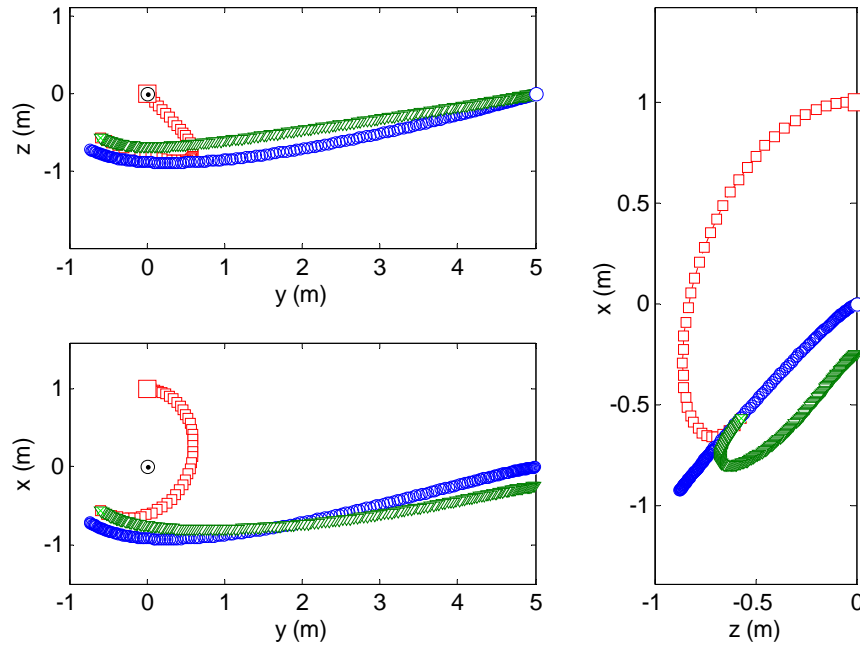


Figure 100. Minimum energy solution (IDVD 3rd): 2D planar projection of the trajectory.

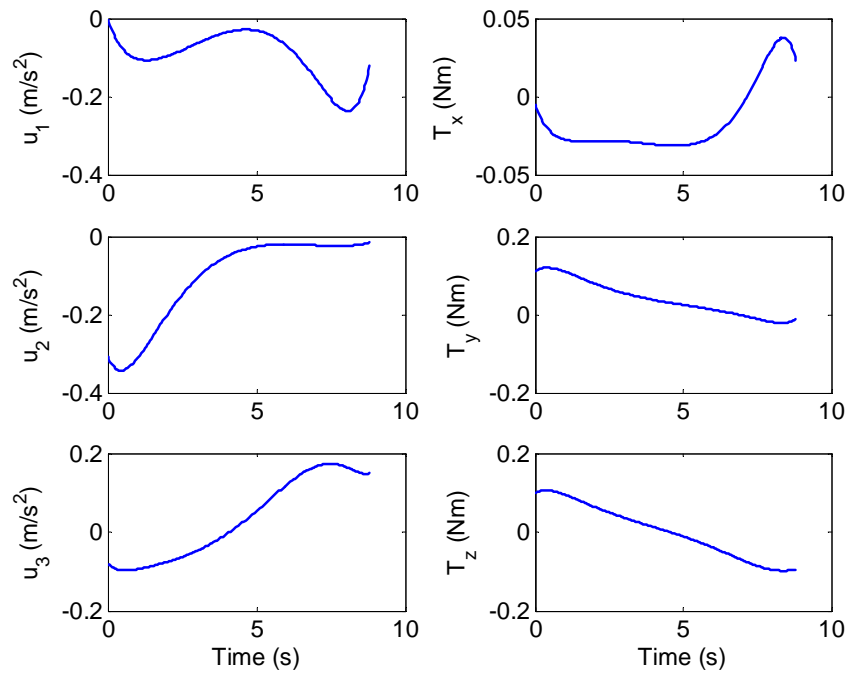


Figure 101. Minimum energy solution (IDVD 3rd): control history.

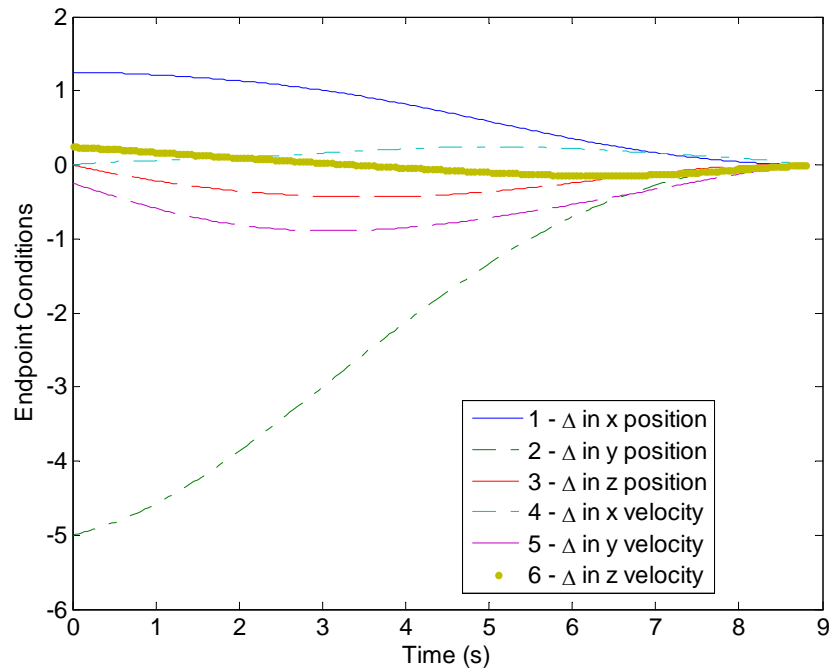


Figure 102. Minimum energy solution (IDVD 3rd): history of the translational endpoint discrepancies.

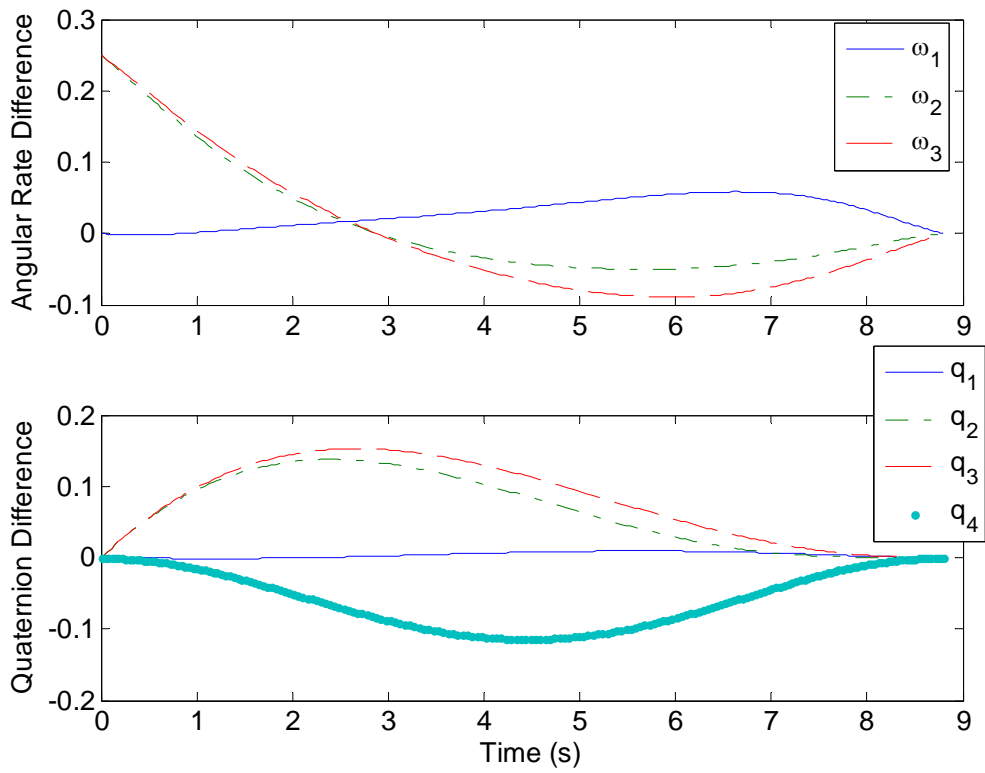


Figure 103. Minimum energy solution (IDVD 3rd): time history of the rotational endpoint discrepancies.

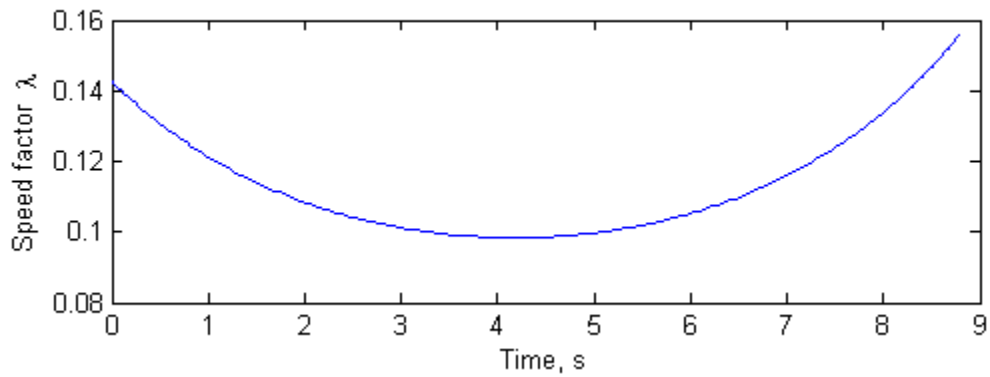


Figure 104. Minimum energy solution (IDVD 3rd): time history of the speed factor.

2. Minimum Time Cost Varying 3rd Parameter Set Results

The final PI based on the minimum time cost is $J = 3.734$ seconds. Figure 105 shows the 3D representation of the trajectory in Figure 106 shows the planar projection of the same trajectory. The control solution is shown in Figure 107, which although is smooth by construction, approaches the bang-bang nature of an optimal solution computed with respect to a minimum time cost function presented in Chapter III.C.1. Figures 108 and 109 show the time history of the endpoint discrepancies and verify that the endpoint conditions were met at the final time. Figure 110 shows the speed factor, which reveals that the chaser is speeding up along the spatial trajectory and then slowing down as it approaches the final desired states. This type of qualitative behavior is inherent to bang-bang, minimum time solutions to general maneuvers which apply max control at the beginning to increase speed of the maneuver, then max control at the end to decrease speed and arrive at the desired final states.

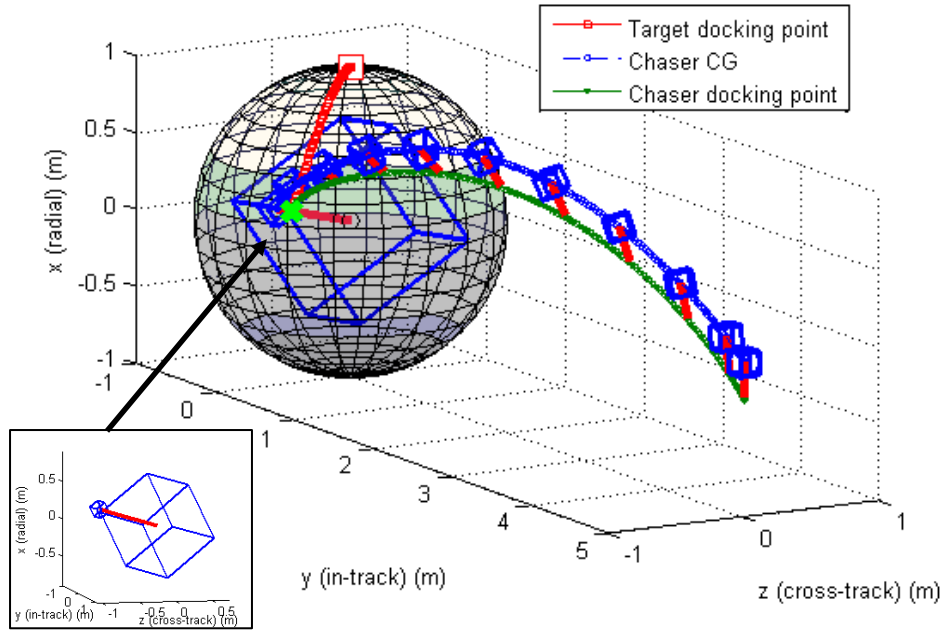


Figure 105. Minimum time solution (IDVD 3rd): the 3D optimal rendezvous trajectory.

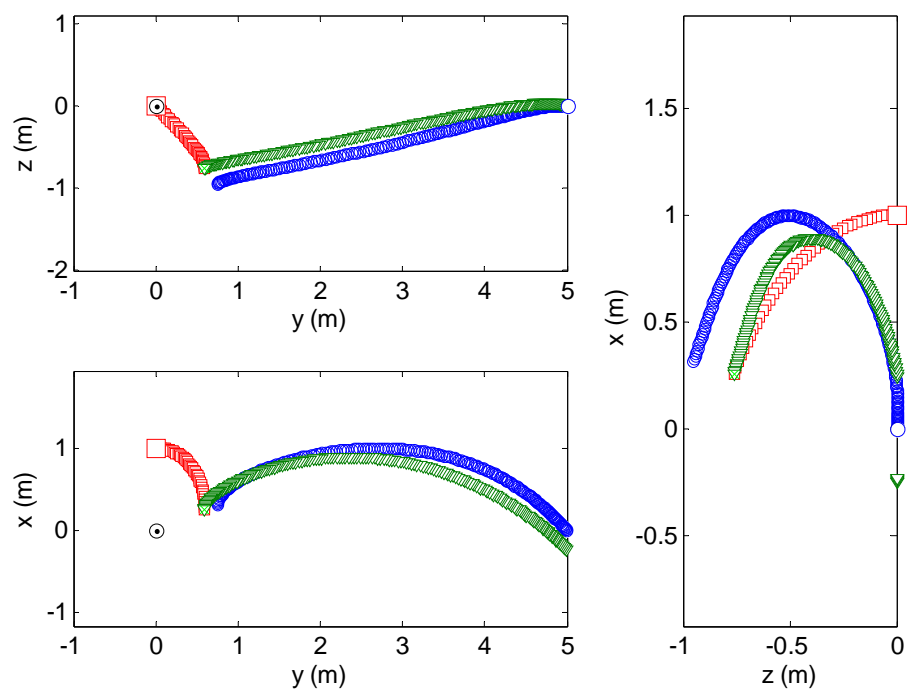


Figure 106. Minimum time solution (IDVD 3rd): 2D planar projection of the trajectory.

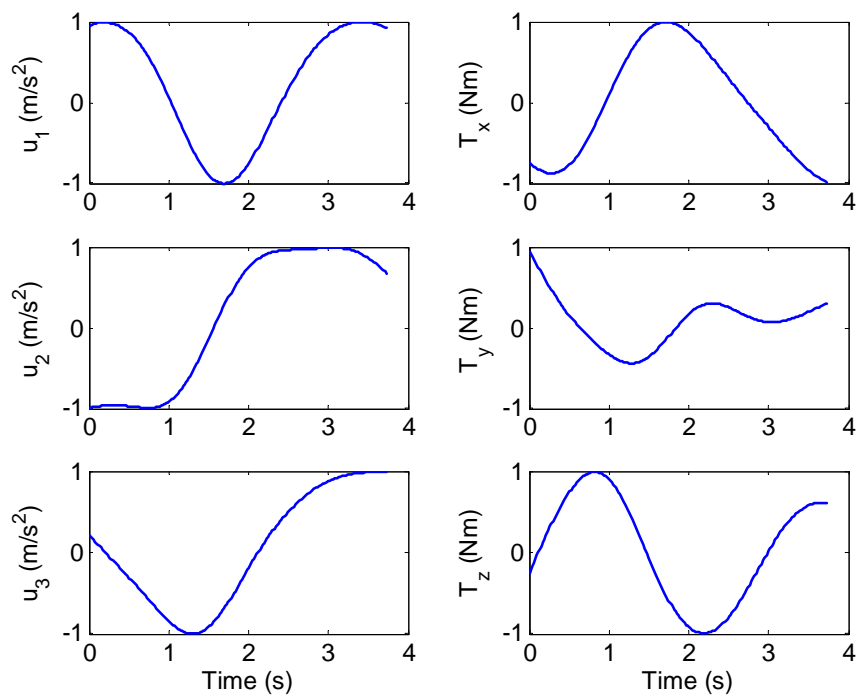


Figure 107. Minimum time solution (IDVD 3rd): control history.

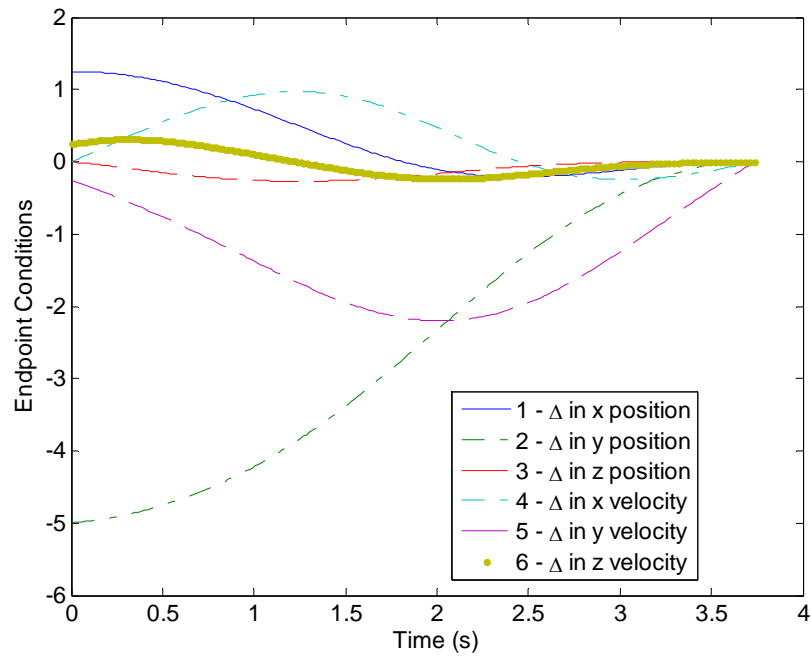


Figure 108. Minimum time solution (IDVD 3rd): history of the translational endpoint discrepancies.

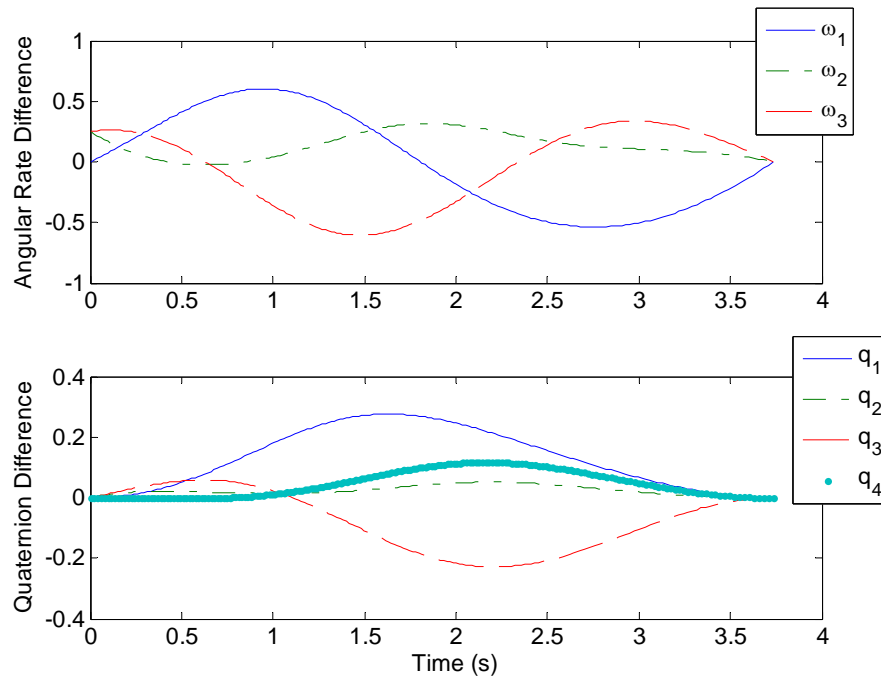


Figure 109. Minimum time solution (IDVD 3rd): history of the rotational endpoint discrepancies.

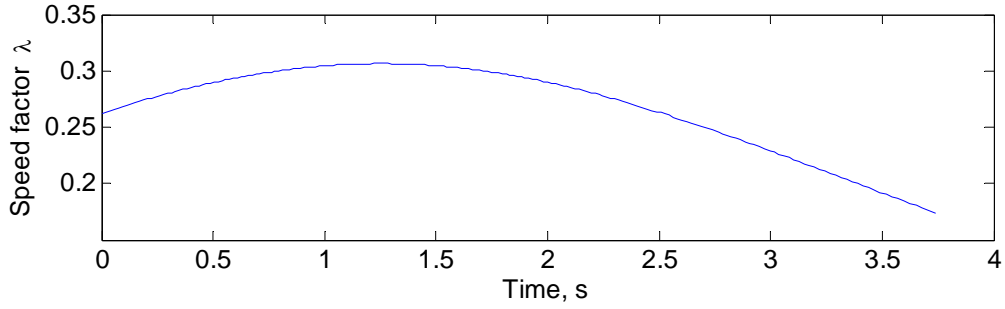


Figure 110. Minimum time solution (IDVD 3rd): time history of the speed factor.

3. Minimum Fuel Cost Varying 3rd Parameter Set Results

The final PI based on the minimum fuel cost is $J = 2.8312$. Figure 111 shows the 3D representation of the trajectory in Figure 112 shows the planar projection of the same trajectory. The control solution is shown in Figure 113. Upon close inspection, the control solution for each component has more action at the beginning and ends of the maneuver, exhibiting qualities of a bang-off-bang structure consistent with the cost function presented in Chapter III.C.1. Figures 114 and 115 show the time history of the endpoint discrepancies and verify that the endpoint conditions were met at the final time. Figure 116 shows the speed factor, which is similar to the minimum energy solution presented in the previous section.

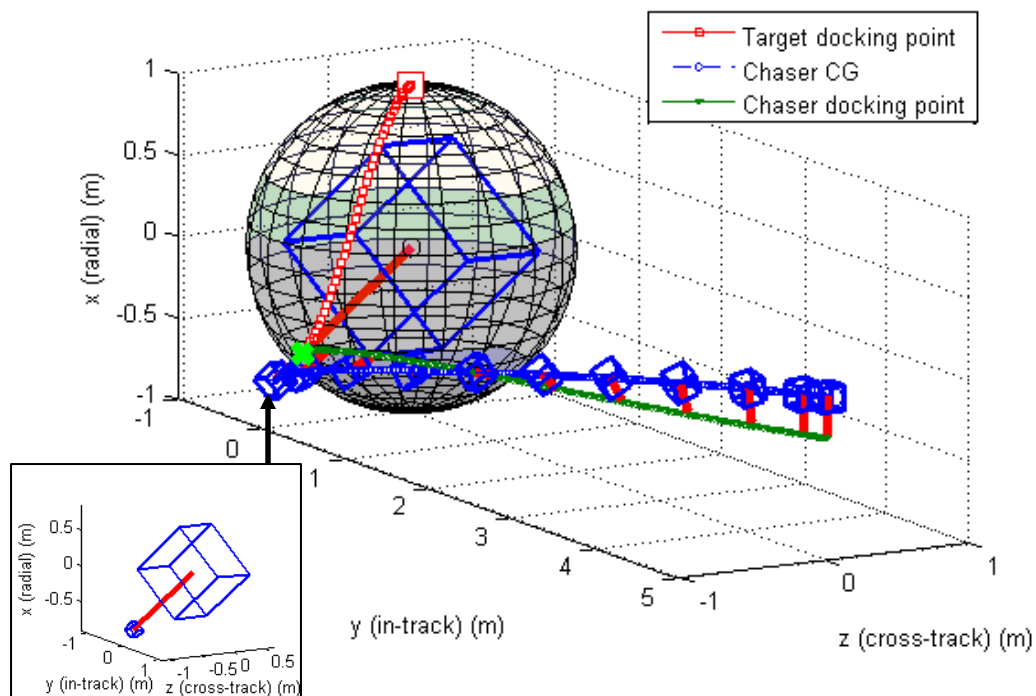


Figure 111. Minimum fuel solution (IDVD 3rd): the 3D optimal rendezvous trajectory.

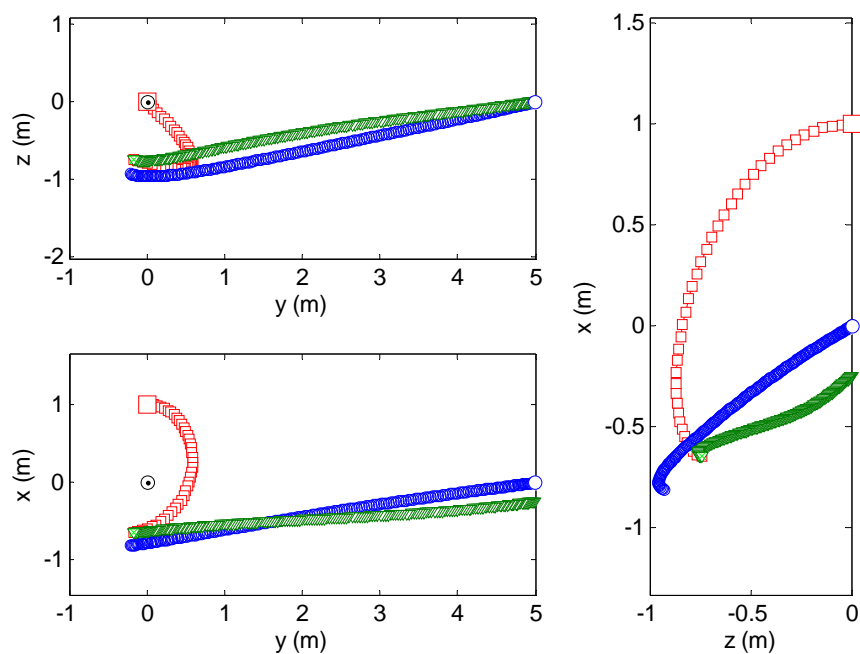


Figure 112. Minimum fuel solution (IDVD 3rd): 2D planar projection of the trajectory.

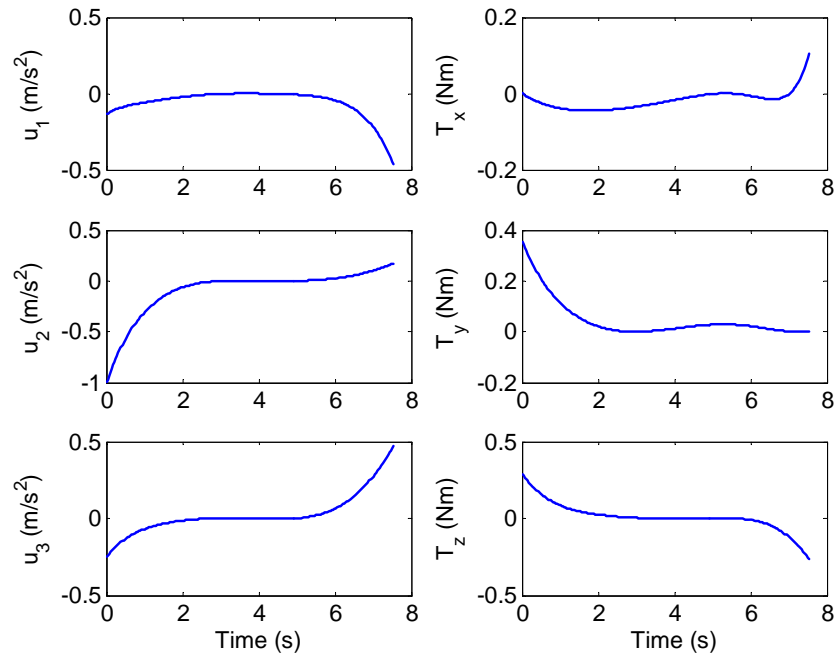


Figure 113. Minimum fuel solution (IDVD 3rd): control history.

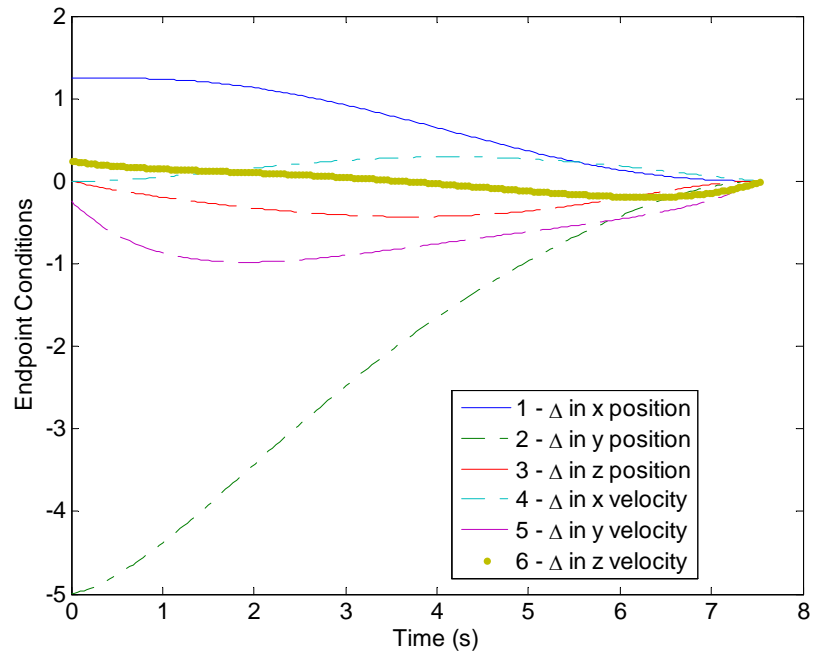


Figure 114. Minimum fuel solution (IDVD 3rd): history of the translational endpoint discrepancies.

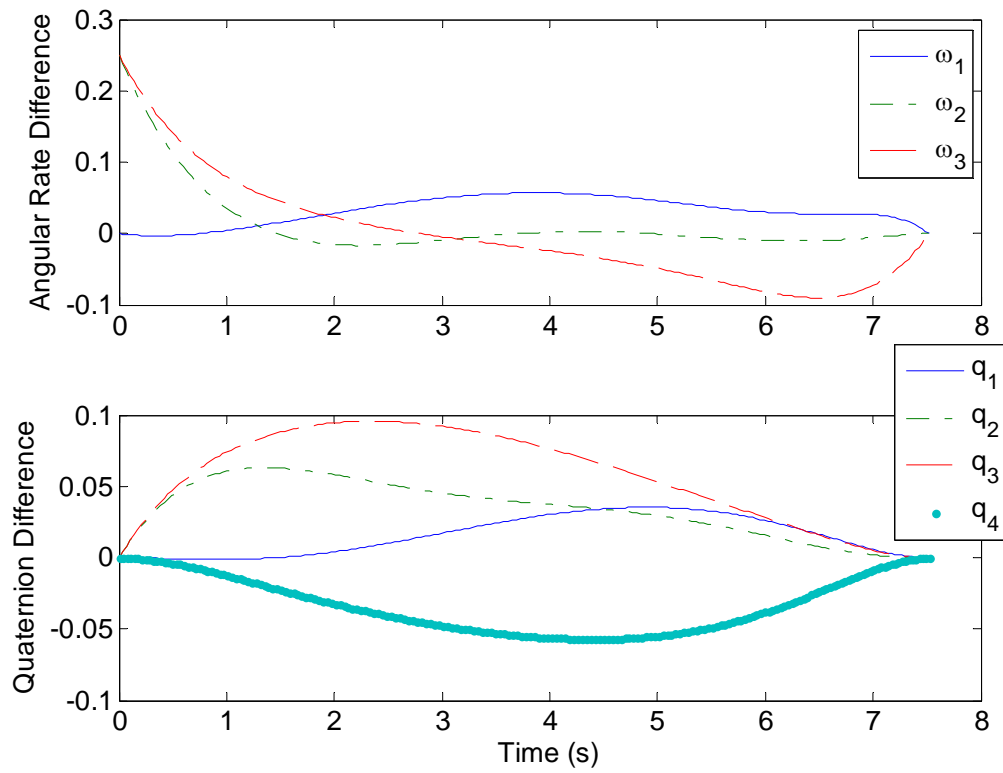


Figure 115. Minimum fuel solution (IDVD 3rd): history of the rotational endpoint discrepancies.

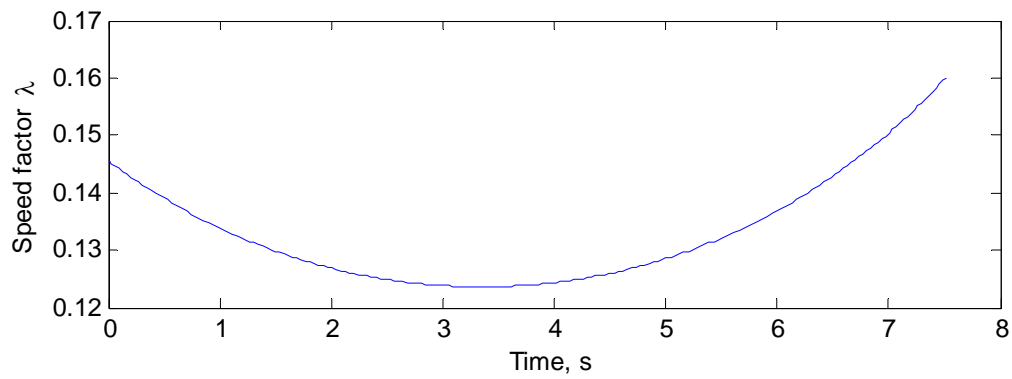


Figure 116. Minimum time solution (IDVD 3rd): time history of the speed factor.

4. Minimum Energy Varying 3rd–4th Parameter Set Results

The problem stated in Chapter V.B.1 is also solved using the IDVD (3rd–4th) method based on the minimum energy cost function from Equation (15). The results are extremely similar to the solution, as shown in the 3-D trajectory of Figure 117, to the same problem formulation based on the IDVD (3rd) method, except the increased flexibility in the polynomial due to the extra basis functions allows for smoother controls at the endpoints, as shown in Figure 118 and later in Figure 123, when compared with IDVD (3rd). The solution took 623 seconds to compute and had a PI of $J = 0.2458$, only a 0.12% reduction in PI. It should be noted that using this formulation, you can specify constraints on the jerk profiles while still using the acceleration profile as your control vector. This attribute is based on the polynomial formulation of the trajectory and cannot be implemented using pseudospectral techniques. To set constraints on jerk using pseudospectral techniques, the control vector would have to be based on jerk and not acceleration.

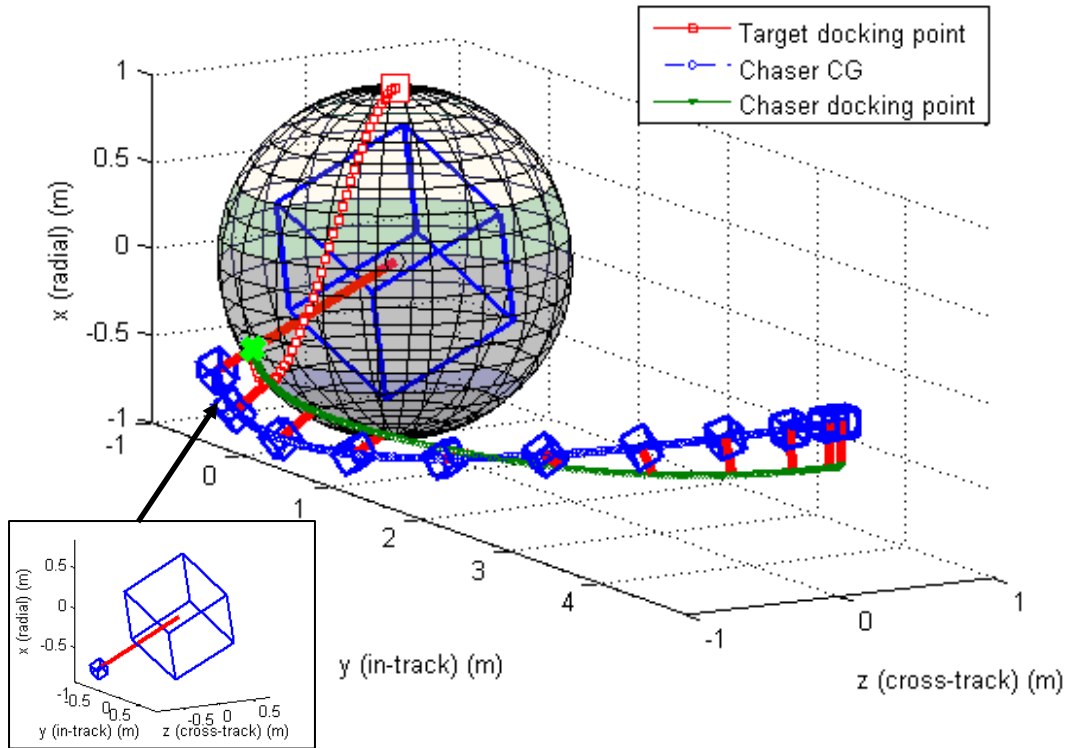


Figure 117. Minimum energy solution (IDVD 3rd–4th): 3D optimal rendezvous trajectory.

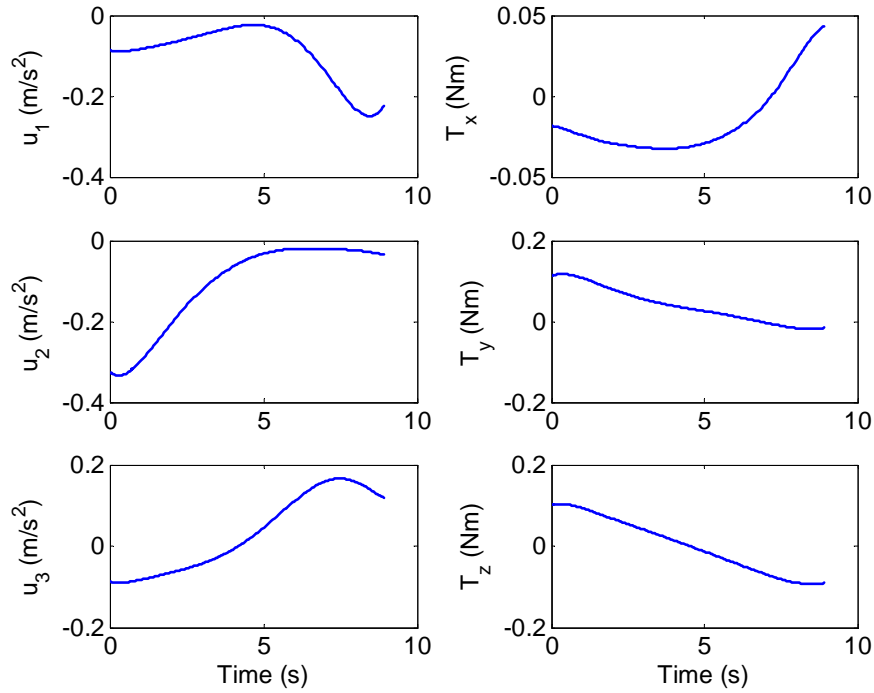


Figure 118. Minimum energy solution (IDVD 3rd–4th): control history.

5. Minimum Time Varying 3rd–4th Parameter Set Results

The minimum time solution using the IDVD (3rd–4th) method is shown in Figure 119. It has a PI of $J = .3.6582$ and took 86.6 seconds to compute. The 3D trajectory looks similar to that obtained by IDVD (3rd), but the extra flexibility in the polynomial provided by an increased set of basis function allows for shaper transitions in the controls, shown in Figure 120, mimicking that of a bang-bang controller. Although the computational cost rose to 86.6 seconds, it is still well below the 8+ hour computational cost of GPOPS for a 200 node representation.

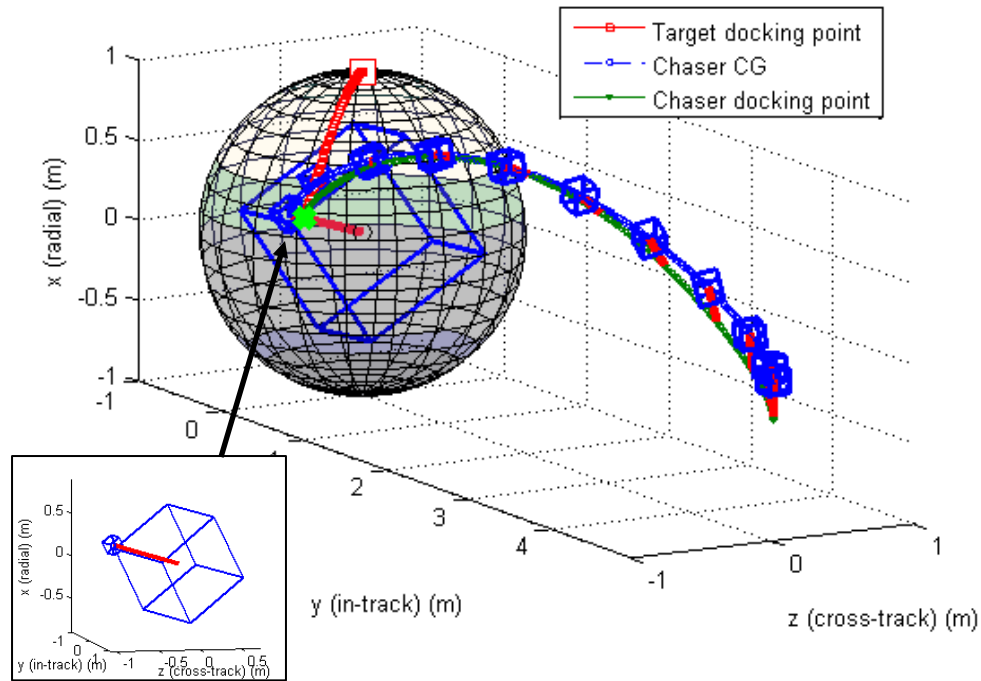


Figure 119. Minimum time solution (IDVD 3rd–4th): 3D optimal rendezvous trajectory.

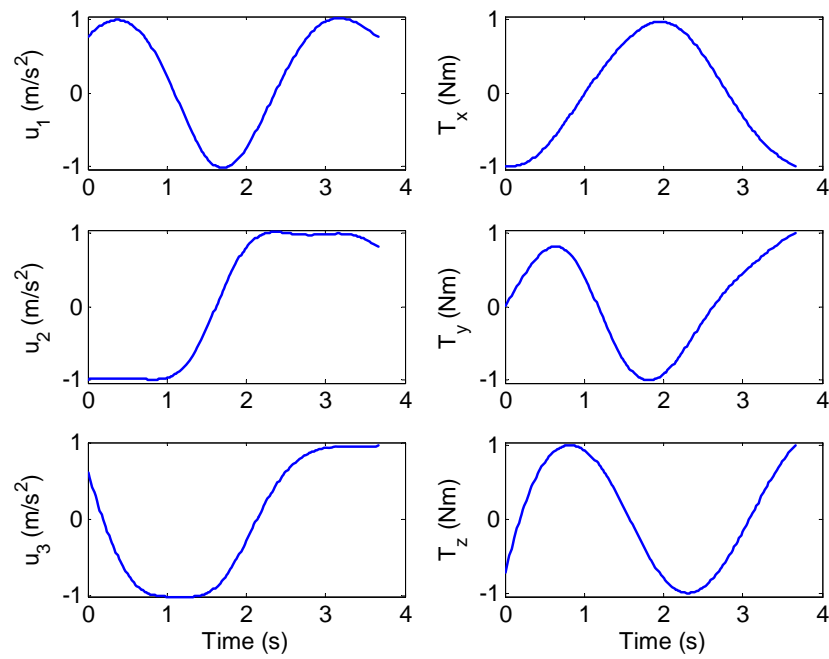


Figure 120. Minimum time solution (IDVD 3rd–4th): control history.

6. Minimum Fuel Varying 3rd–4th Parameter Set Results

The minimum fuel solution obtained using the IDVD (3rd–4th) method is shown in Figure 121. Again, the trajectory is very similar to the solution calculated from the IDVD (3rd) except the increased flexibility allows for shaper transitions in control at the endpoints. The bang-off-bang type of control is evident from examining the control histories in Figure 122. The resulting PI is $J = 2.7552$, taking 405.5 seconds to compute.

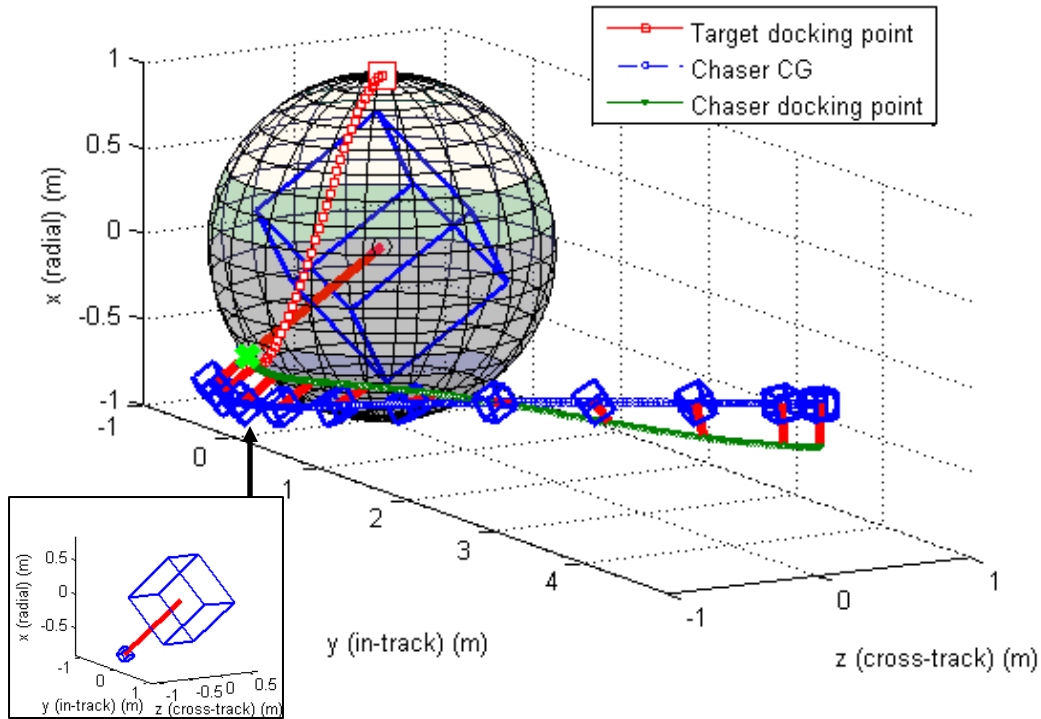


Figure 121. Minimum fuel solution (IDVD 3rd–4th): 3D optimal rendezvous trajectory.

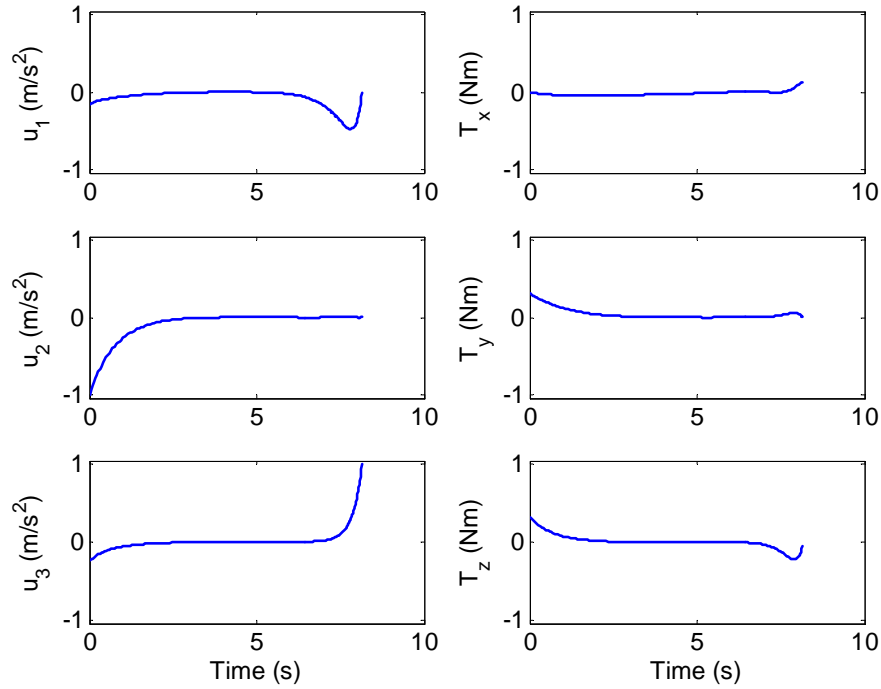


Figure 122. Minimum fuel solution (IDVD 3rd–4th): control history.

7. Propagated IDVD Solutions

As with the solutions obtained by the pseudospectral method in Chapters II and III, the solutions for the controls obtained with the IDVD method were propagated using the same mechanism as the pseudospectral method stated in Chapter III.C.4. The resulting discrepancies in the endpoints resulting from all three cost functions using the IDVD (3rd) method are displayed in Tables 23–25, with minimum time plotted in Figures 123–124.

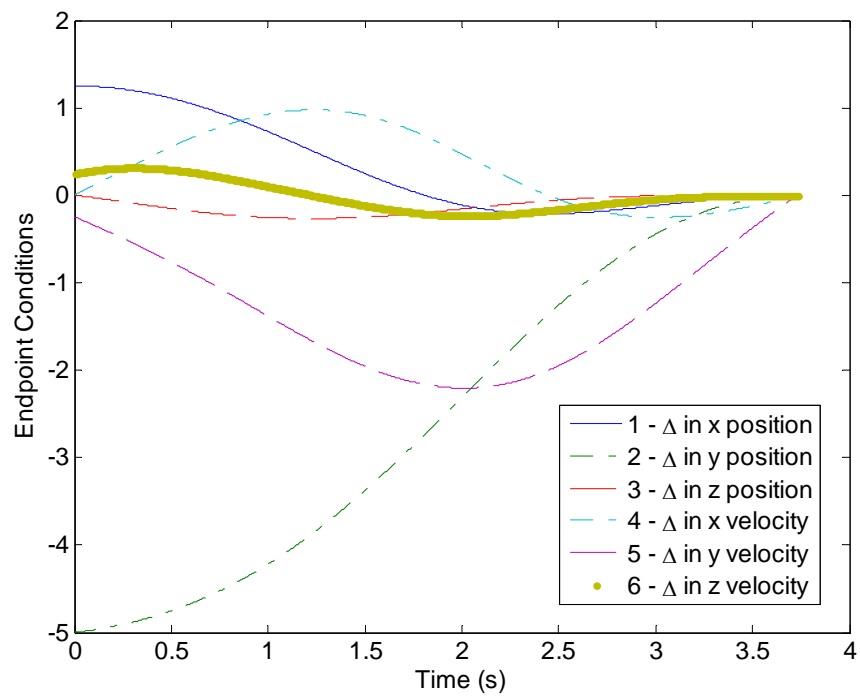


Figure 123. Translational discrepancies for propagated IDVD (3rd) minimum time solution.

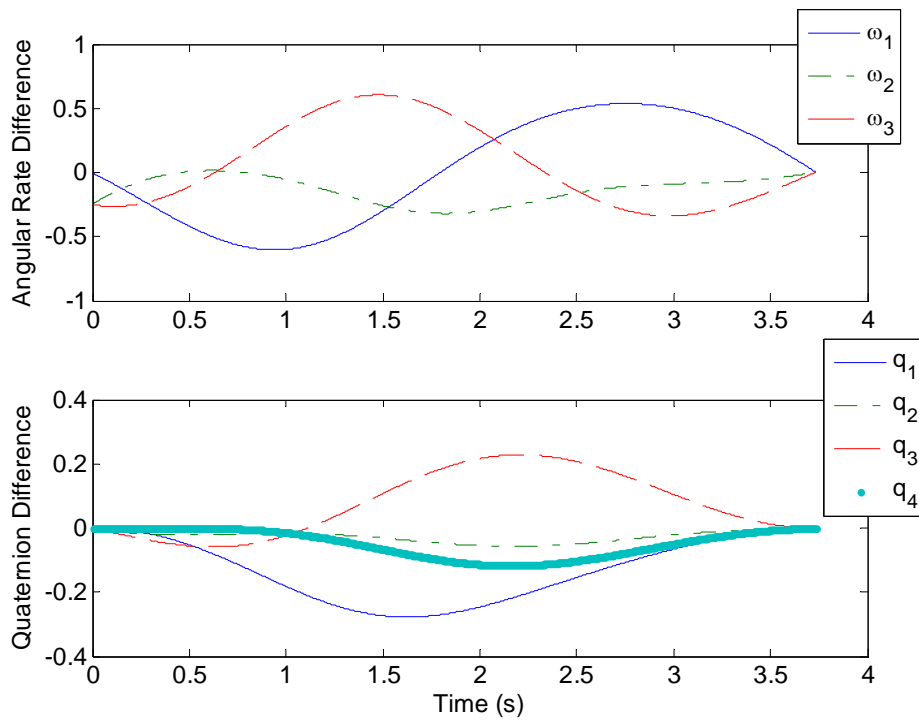


Figure 124. Rotational discrepancies for propagated IDVD (3rd) minimum time solution.

e	<i>Resulting Value</i>	e	<i>Resulting Value</i>	e	<i>Resulting Value</i>
$e_1(t_f)$	0.0006	$e_8(t_f)$	0.0013	$\zeta_2(t_f)$	N/A
$e_2(t_f)$	-0.0002	$e_9(t_f)$	-0.0001	$\zeta_3(t_f)$	N/A
$e_3(t_f)$	0.0002	$e_{10}(t_f)$	-0.0009	$\zeta_4(t_f)$	N/A
$e_4(t_f)$	-0.0009	$e_{11}(t_f)$	-0.0003	$\zeta_5(t_f)$	N/A
$e_5(t_f)$	0.0004	$e_{12}(t_f)$	0.0001	$\zeta_6(t_f)$	N/A
$e_6(t_f)$	-0.0009	$e_{13}(t_f)$	-1.0211e-005	$\zeta_7(t_f)$	N/A
$e_7(t_f)$	0.0055	$\zeta_1(t_f)$	N/A	$H(t_f)$	N/A

Table 23. Value of terminal conditions at the final time for IDVD (3rd) minimum time solution.

e	<i>Resulting Value</i>	e	<i>Resulting Value</i>	e	<i>Resulting Value</i>
$e_1(t_f)$	0.0041	$e_8(t_f)$	0.0003	$\zeta_2(t_f)$	N/A
$e_2(t_f)$	-0.0050	$e_9(t_f)$	1.4198e-006	$\zeta_3(t_f)$	N/A
$e_3(t_f)$	-0.0023	$e_{10}(t_f)$	-0.0020	$\zeta_4(t_f)$	N/A
$e_4(t_f)$	-0.0029	$e_{11}(t_f)$	-0.0001	$\zeta_5(t_f)$	N/A
$e_5(t_f)$	0.0021	$e_{12}(t_f)$	-0.0015	$\zeta_6(t_f)$	N/A
$e_6(t_f)$	-0.0002	$e_{13}(t_f)$	-0.0007	$\zeta_7(t_f)$	N/A
$e_7(t_f)$	0.0016	$\zeta_1(t_f)$	N/A	$H(t_f)$	N/A

Table 24. Value of terminal conditions at the final time for IDVD (3rd) minimum energy solution.

e	<i>Resulting Value</i>	e	<i>Resulting Value</i>	e	<i>Resulting Value</i>
$e_1(t_f)$	0.0019	$e_8(t_f)$	0.0017	$\zeta_2(t_f)$	N/A
$e_2(t_f)$	-0.0011	$e_9(t_f)$	2.6406e-005	$\zeta_3(t_f)$	N/A
$e_3(t_f)$	-0.0002	$e_{10}(t_f)$	-0.0026	$\zeta_4(t_f)$	N/A
$e_4(t_f)$	-0.0027	$e_{11}(t_f)$	-0.0004	$\zeta_5(t_f)$	N/A
$e_5(t_f)$	0.0014	$e_{12}(t_f)$	-0.0005	$\zeta_6(t_f)$	N/A
$e_6(t_f)$	-0.0008	$e_{13}(t_f)$	-0.0011	$\zeta_7(t_f)$	N/A
$e_7(t_f)$	0.0036	$\zeta_1(t_f)$	N/A	$H(t_f)$	N/A

Table 25. Value of terminal conditions at the final time for IDVD (3rd) minimum fuel solution.

8. Summary of Close Approach Results for Baseline Maneuver

Results for different variations of the IDVD theme pertaining to the baseline maneuver introduced in Chapter III and discussed throughout this dissertation are summarized in Table 26.

	GPOPS			IDVD (Euler Angles) (vary endpoint accelerations)		
cost	Min Time	Min Energy	Min Fuel	Min Time	Min Energy	Min Fuel
Final Time	3.5086	8.8943	8.1182	3.7775	10.0000	9.0792
Energy	10.4471	0.2445	1.1587	5.0869	0.3495	0.5738
Fuel	20.9419	3.6941	2.4863	13.3627	4.8462	3.9329
CPU Time	33,370.1	86,040.7	84,261.6	10.6	6.8	11.0

	IDVD (quaternions) (vary endpoint accelerations)			IDVD (quaternions) (vary endpoint jerk/accelerations)		
cost	Min Time	Min Energy	Min Fuel	Min Time	Min Energy	Min Fuel
Final Time	3.7341	8.8868	7.5181	3.6582	8.9137	8.1442
Energy	5.2641	0.2461	0.3875	6.2309	0.2458	0.4131
Fuel	13.4202	3.7177	2.8312	15.0982	3.7044	2.7552
CPU Time (sec)	11.5	16.1	32.1	86.6	623.0	405.5

Table 26. Summary of performance indices and computational time for a variety of trajectory generation methods.

For the cases of GPOPS and IDVD employing quaternions, Figure 125 shows the resulting minimum energy control histories represented in a single plot. Both the control histories calculated using IDVD match well with the solution provided by GPOPS. The similarities of the solutions are further reinforced by the small deviation in PI with the 3rd derivative IDVD solution, IDVD (3rd), having a PI within 0.65% of the GPOPS solution and the IDVD (3rd–4th) solution, IDVD (3rd–4th), falling within 0.54%.

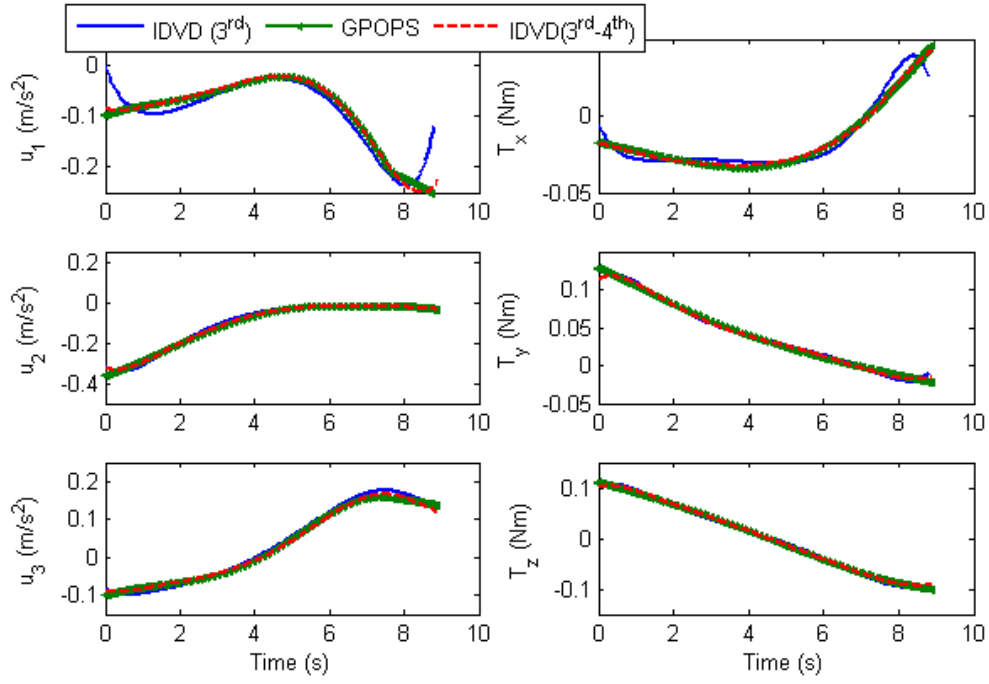


Figure 125. Control history of different methods for the minimum quadratic-control problem formulation.

The overlaid control histories for the minimum time case are shown in Figure 126.

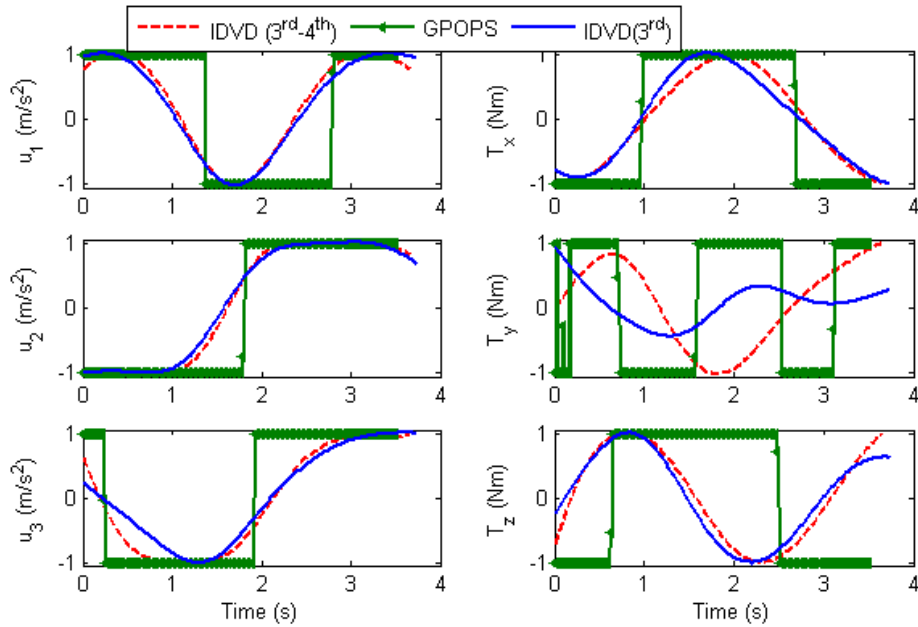


Figure 126. Control history of different methods for the minimum time problem formulation.

For the minimum time case, the reduced performance of the IDVD solutions is mainly due to the inability to implement a bang-bang control strategy. This is because the IDVD results in a smooth, continuous function by construction. Although the IDVD (3rd–4th) strategy is more flexible, having the added capability to modify the endpoint jerk values, it still cannot implement the discontinuous switch of control values inherent to a minimum time solution for this type of problem and shows only minor improvement over the IDVD (3rd) strategy. Although, previous work calculating bang-bang solutions needed to generate an extra control of layer in order to have the required smooth control history for interpolation (Hurni 2009). The PI of the IDVD (3rd) solution was within 6.5% of the GPOPS solution and the IDVD (3rd–4th) solution was within 4.3%.

The minimum fuel solutions are shown in Figure 127. Although the IDVD solutions have a slightly higher PI, they avoid the rapid switching characteristics portrayed by the GPOPS solution inherent to the bang-off-bang control nature associated with the cost function described by Equations (59) and (62). The solution provided by IDVD is also smooth through a region that GPOPS provides a highly oscillatory solution incapable of being implemented. The trade off, of course, is the slightly higher PI associated with the solution, 13.9% increase for IDVD (3rd) and 10.8% for IDVD (3rd–4th).

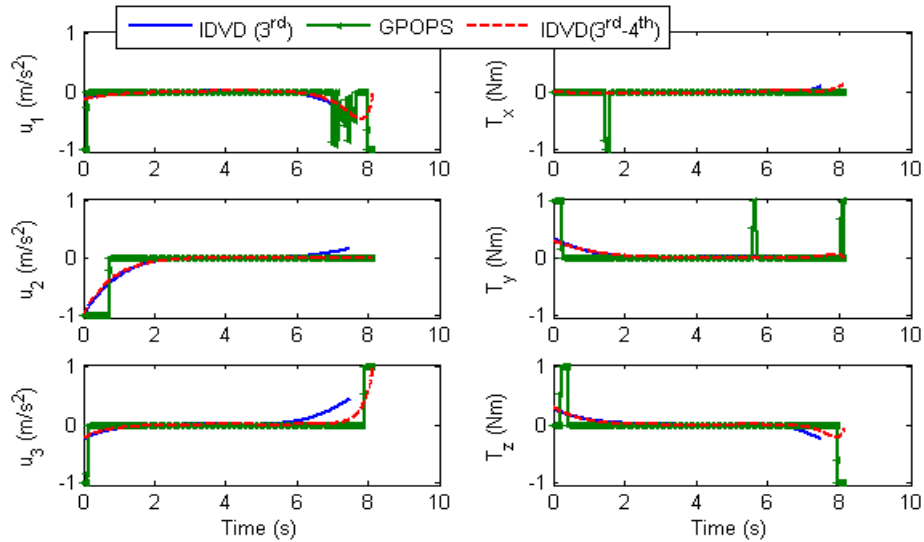


Figure 127. Control history of different methods for the minimum fuel problem formulation.

VI. FURTHER SIMULATIONS AND ANALYSIS

This chapter analyzes the rendezvous final approach problem in several different ways. First, the methods developed in this dissertation are applied to previously published examples. Concepts for closed-loop, real time operations are derived and demonstrated in simulation. Finally, the Inverse Dynamics in the Virtual Domain method is applied to a variety of scenarios with different initial conditions.

A. COMPARISON WITH PREVIOUS STUDIES

1. Problem Formulation in Two Dimensions

Simulations were run to compare results with those published in previous literature. Specifically, the planar case of Ma (Ma et al. 2007) that is based on the Sakawa Shindo (SS) algorithm (Sakawa and Shindo 1980). This article attempts to solve the planar motion minimum time problem of a spacecraft matching position and orientation with a docking point that is rotating. The state vector is defined as:

$$\mathbf{x}_r = [x_r \quad y_r \quad \theta_r \quad v_{xr} \quad v_{yr} \quad \omega_r]^T \quad (147)$$

The subscript r indicates the relative motion of the chaser with respect to the RSO, observed from the targets body fixed frame. There are two force controls u_1 and u_2 , respectively fixed in the x and y chaser spacecraft body frame and one control torque u_3 fixed along the z axis. The controls are bounded such that: $-1 \leq \mathbf{u} \leq 1$. The initial and final conditions are given as:

$$\begin{aligned} \mathbf{x}_{r0} &= (10 \quad 10 \quad \pi/2 \quad 1 \quad 1 \quad 1)^T \\ \mathbf{x}_{rf} &= (1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0)^T \end{aligned} \quad (148)$$

Since the algorithm needs a final time, t_f , as in input, the method is to solve a minimum fuel problem, reducing t_f at each iteration. This iteration of iterations method can be computationally expensive, and the inability to find a solution using the SS algorithm does not guarantee that one does not exist. Furthermore, relative motion due to

the rotating reference frame (not inertially fixed) and the mean motion due to Hill's equation are neglected. For more information on the SS algorithm, the reader is directed to (Sakawa and Shindo 1980; Sakawa 1999). Although the algorithm claims to satisfy Pontryagin's Minimum Principle at every iteration, no information about the costates or transversality conditions are supplied. The final minimum time solution arrived at by this method is 8.14 seconds using 200 nodes for numerical integration by the Heun method (Chyba, Leonard and Sontag 2000). The computation time was claimed to take several minutes, but not stated. Instead, the author stated that the computational time was not the focus of the research, only the computation of a feasible solution.

2. Direct Method (GPOPS) Formulation and Results

The same problem is solved by restricting the GPOPS and IDVD formulations derived in previous sections to two dimensions. The resulting solution for GPOPS provided a minimum time for the maneuver to be 7.6695 seconds. The resulting plot of chaser x and y coordinates fixed are presented in the RSO body frame, as in Ma (2007), is shown in Figure 128. The control histories are shown in Figure 129, as well as the switching conditions developed in Chapter III.B. The controls obey the bang-bang nature stated in Ma (2007) and Chapter III of this dissertation, which are dictated by the switching functions. The controls presented in Ma are said to approach the bang-bang structure but appear rather smooth. Figures 130 and 131 show the time history of the endpoint conditions, the difference in position and velocity of the docking points, as well as angular rate and orientation of the vehicles, converging to zero. The transversality conditions and Hamiltonian shown in Figures 132 and 133 further reinforce the optimal nature of the control for the minimum time cost. Still, the major drawback of this formulation and method is that the calculation time of the solution was 1,416.2 seconds (23.6 minutes) for only a 50 node solution (not shown), and the calculated solution for 200 nodes, shown in Figures 128–133, required 40,659.0 seconds of computational time.

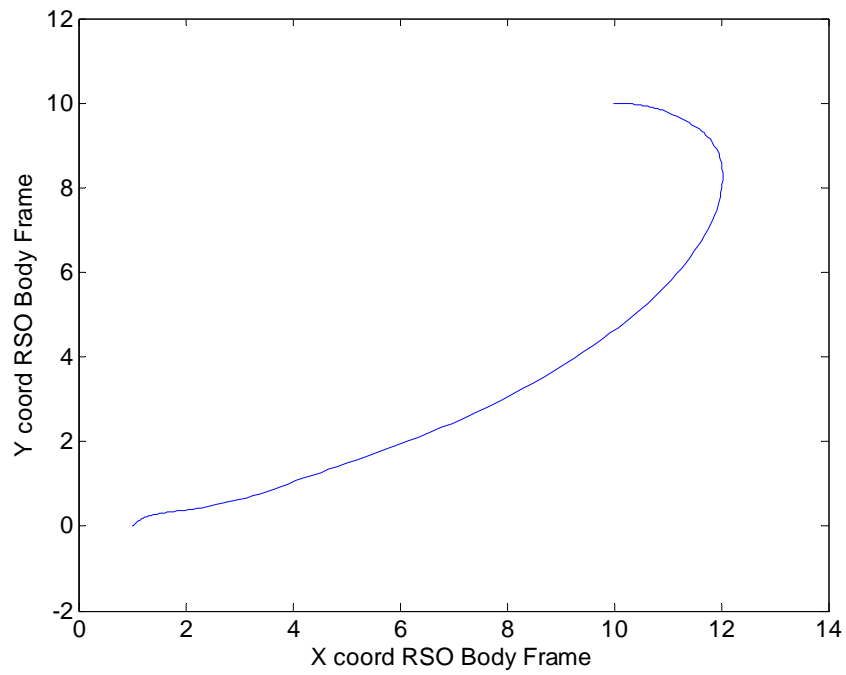


Figure 128. 2D Case (GPOPS): optimal minimum time trajectory of the chaser in the RSO body frame.

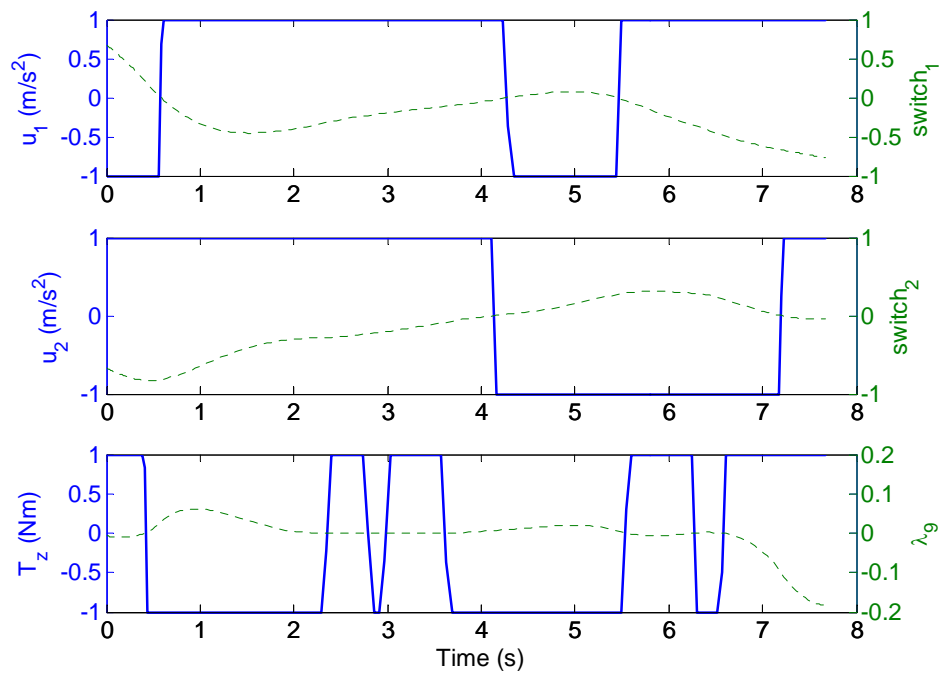


Figure 129. 2D Case (GPOPS): optimal minimum time control history.

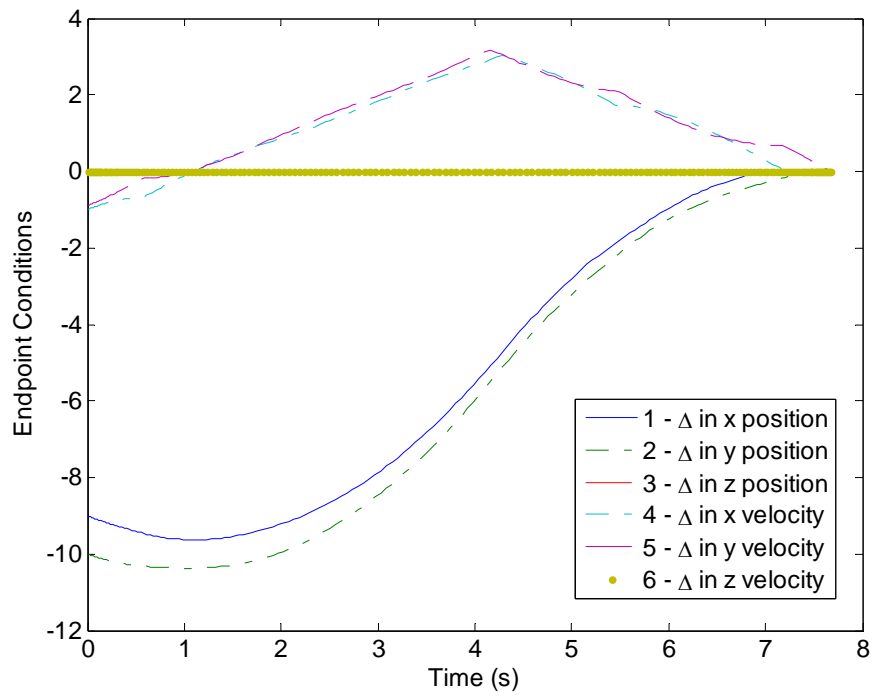


Figure 130. 2D Case (GPOPS): history of the translational endpoint conditions.

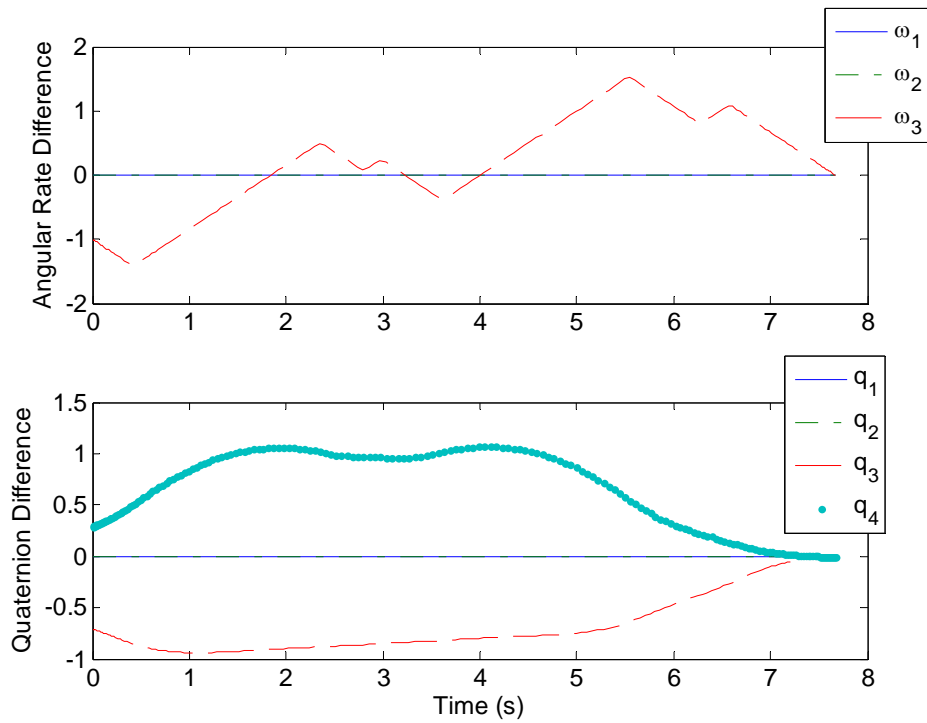


Figure 131. 2D Case (GPOPS): history of the attitude endpoint conditions.

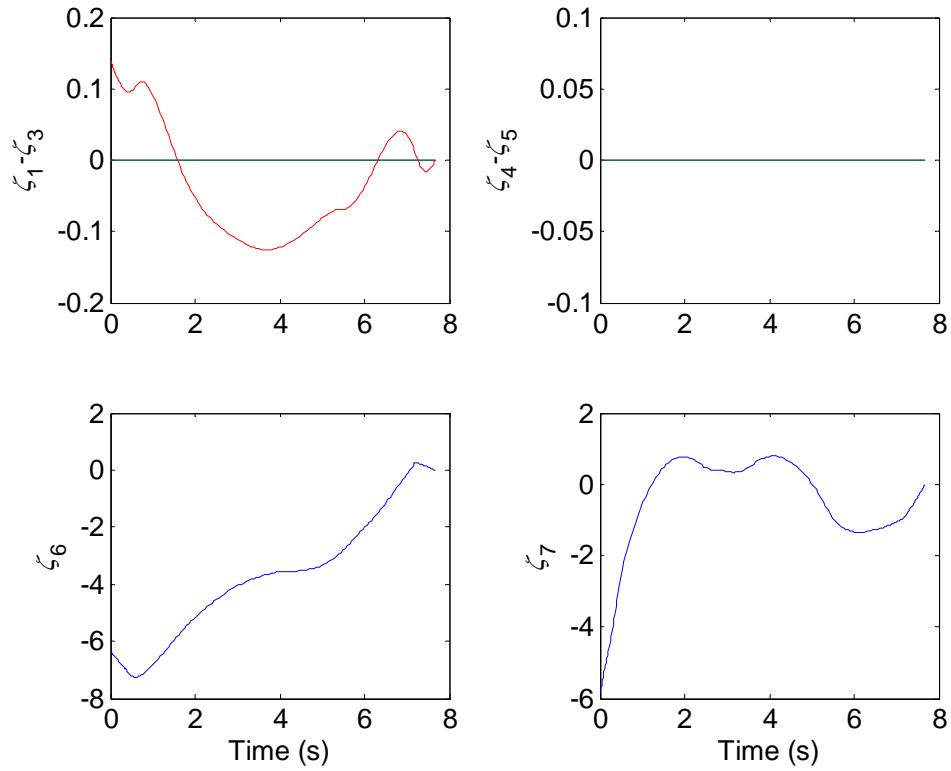


Figure 132. 2D Case (GPOPS): history of the transversality conditions.

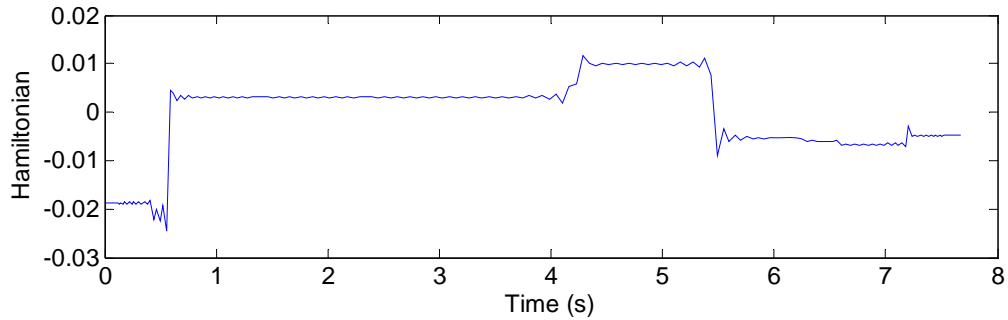


Figure 133. 2D Case (GPOPS): history of the Hamiltonian.

For comparison purposes, the GPOPS method was run with a minimum fuel cost based on Equation (93) but with a multiplier of 0.005 (Ma et al. 2007) and a maximum final time of 8.14 seconds. The solution took 1,953.1 seconds (32.6 minutes) to calculate a 50 node solution and provided a PI of 0.0574 (compared to the 0.0788 PI result from the SS based method).

3. IDVD Formulation and Result

The IDVD method was also used to calculate the trajectory based, shown in Figure 134, on the 2D problem posed by Ma (Ma et al. 2007). The associated control history is shown in Figure 135. The endpoint conditions are shown in Figures 136 and 137. Again, the same formulation and methodology was used as in Chapter V.A, varying acceleration at the endpoints, but the problem was constrained to planar motion. The resulting costs and plots were generated using 200 points (the resulting IDVD is analytic, so any number of nodes or waypoints can be used to present the final solution). This final time calculated, 8.7525 seconds, is higher than both the previous methods, but the solution only took 10.3 seconds to compute. When the IDVD (3rd–4th) method is used, the final time to complete the maneuver is decreased to 8.2086 seconds, but the computational time to arrive at the solution increase to 214.7 seconds.

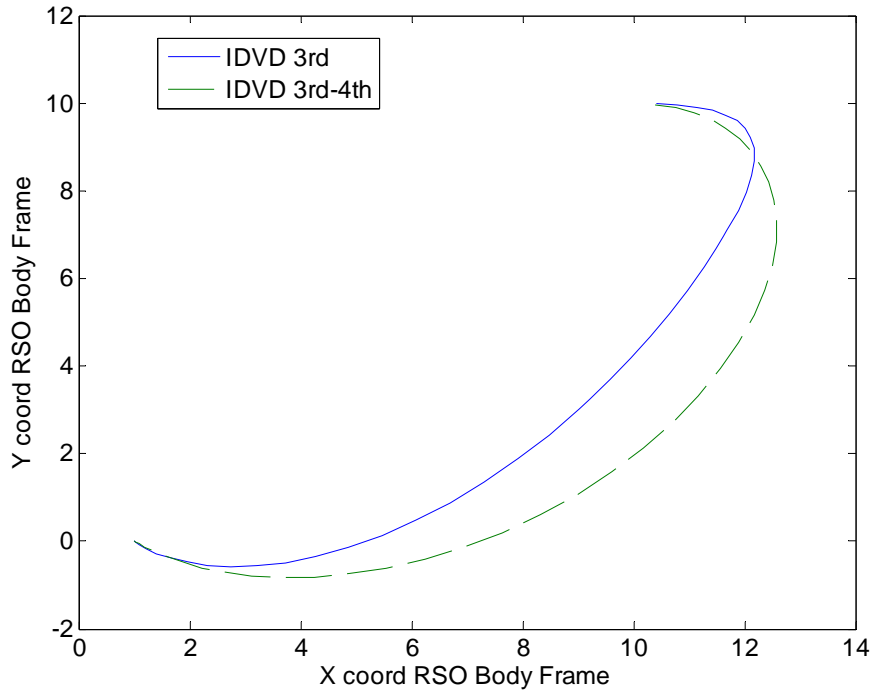


Figure 134. 2D Case (IDVD): minimum time trajectory of the chaser in the RSO body frame.

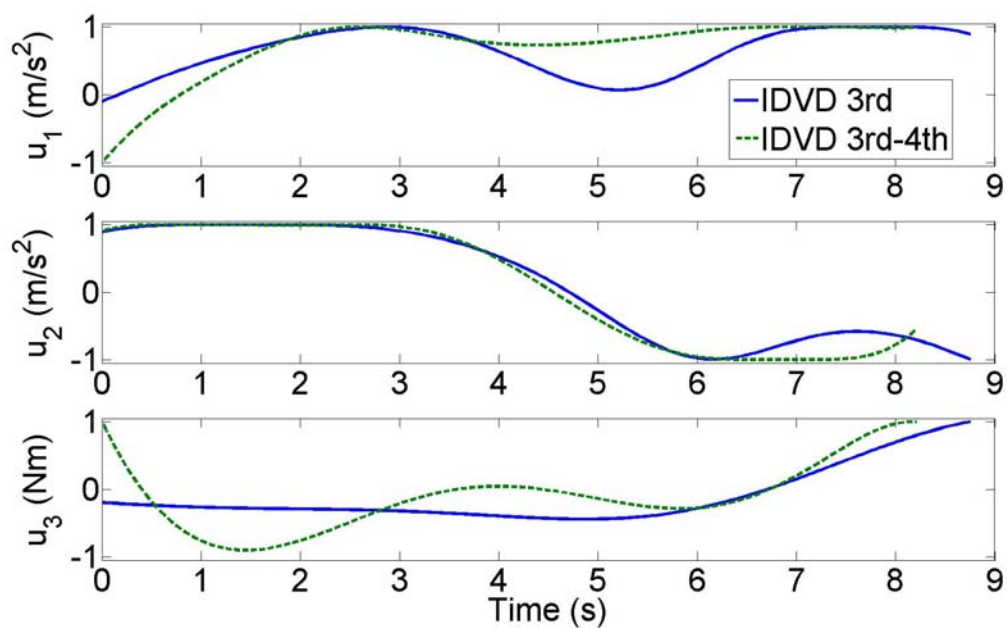


Figure 135. 2D Case (IDVD 3rd): minimum time control history.

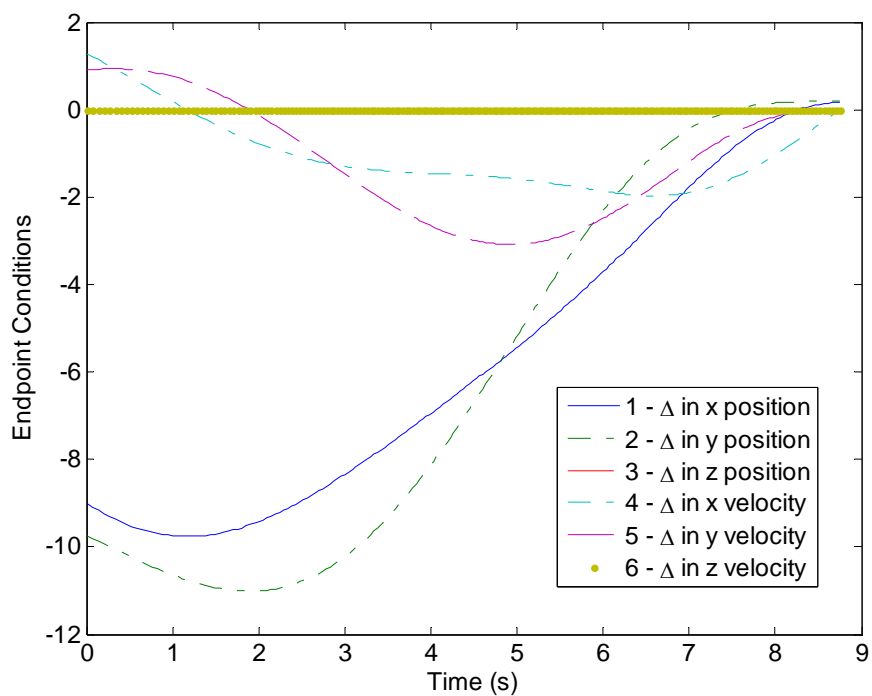


Figure 136. 2D Case (IDVD 3rd): history of the translational endpoint conditions.

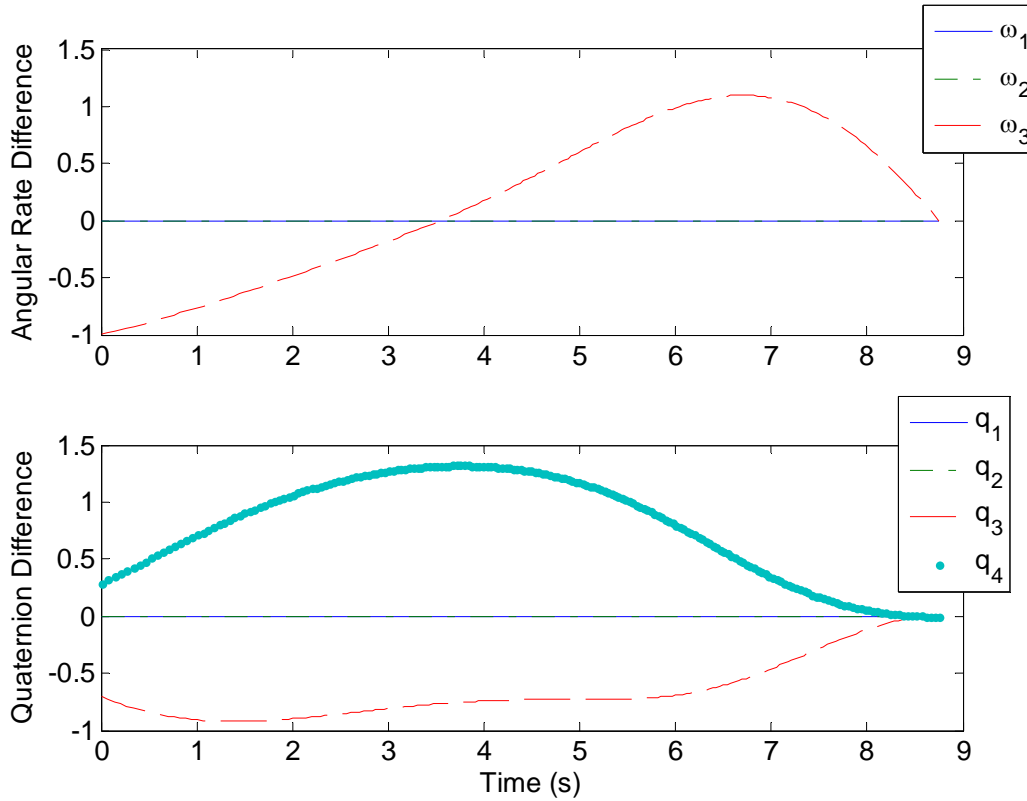


Figure 137. 2D Case (IDVD 3rd): history of the attitude endpoint conditions.

The increased flexibility of the IDVD (3rd–4th) basis functions permits the solution to behave more like the bang-bang structure of the optimal solution, allowing for a reduced final time for the maneuver to complete.

4. Comparisons of Trajectory Generation Methods for 2D Example

Table 27 shows a comparison between methods for the minimum time solution. 200 nodes were chosen for the calculation and a factor of 0.005 was multiplied by the minimum fuel cost to be consistent with previous literature (Ma 2007). Although IDVD (3rd) has a greater final maneuver time, the computational time is drastically reduced. The computational time, simply stated as several minutes, and final minimum energy cost of the SS solution presented by Ma (2007), were not reported. The solution providing the best minimum time cost was the bang-bang solution solved for by GPOPS, but the solution came at an extensive computational price.

	GPOPS	IDVD (3rd)	IDVD (3rd-4th)	Ma (SS)
cost	Min Time	Min Time	Min Time	Min Time
Final Time	7.6695	8.7525	8.2086	8.1400
Energy	11.3266	5.9307	6.9566	N/A
Fuel*	0.1135	0.0779	0.0824	0.0788
CPU Time (sec)	40,659.0	10.3	214.7	N/A

Table 27. Summary of performance indices and computational time for a 2D scenario. *A factor of 0.005 was multiplied by the minimum fuel cost to be consistent with the formulation presented by Ma (2007).

B. CYCLICAL NATURE OF PROBLEM SOLUTION PERFORMANCE INDEX

The performance index of rendezvous of a spacecraft with a tumbling object exhibits a cyclical nature as seen by Figures 138 and 139 for two scenarios. Figure 138 shows the cyclic nature of the PI with respect to a time value associated with waiting to start the maneuver. If the initial angular velocity of the RSO occurs around a principal moment of inertia, then the motion of the docking point is periodic, containing only circular motion in the plane perpendicular to the angular velocity vector. A simple example of fixing $\lambda = 1$ for the entire maneuver and then varying the wait time before maneuver start (in essence is the same as varying the initial orientation of the RSO), shows that the PI associated with completing the maneuver is periodic with time. Figure 139 shows the cyclic cost nature for the case of the docking point motion, not being constrained to a plane perpendicular to the initial angular velocity vector of the RSO. Such is the case of the RSO having a nonidentity inertia matrix and initial angular velocity vector that is not coincident with a principal moment of inertia, the PI is cyclic, but does not repeat. For this case, the time value shown is the total time of the maneuver. For both cases, there is an inherent cyclical nature to the PI pertaining either the wait time until commencing the maneuver or the total time of the maneuver. This added complexity demonstrates the increased influence of the maximum final time on the solution. For any given circumstance, allowing more time to perform the maneuver may not necessarily decrease the PI for minimum fuel or minimum energy maneuvers as would be the case if rendezvous was to a fixed point in space. For this reason, initial

guesses should be chosen judiciously and when possible, as stated in Chapter V.B, and should provide a feasible solution. This way, since the kinematic and dynamic equations are always satisfied though IDVD, there is always a solution to implement at any given time should the user want to terminate the optimization routine.

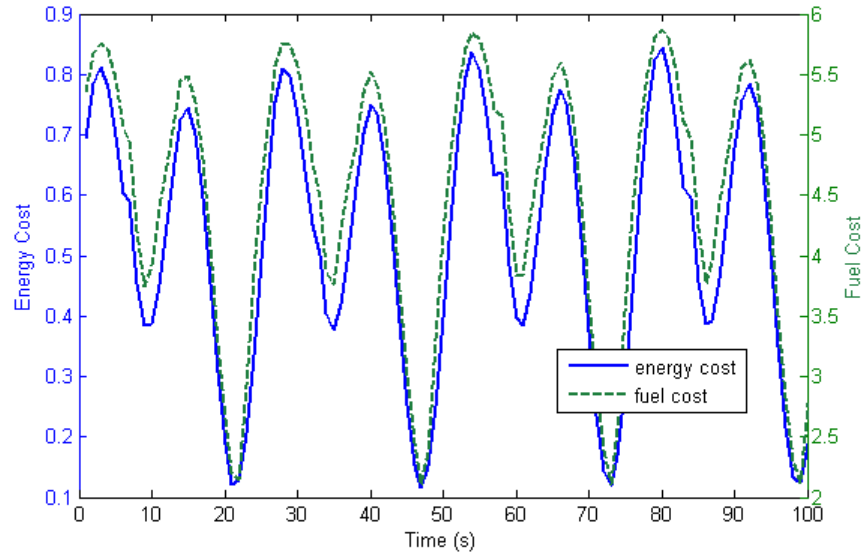


Figure 138. Energy and fuel costs for rendezvous with a tumbling RSO with symmetric inertia matrix. The time shown is a wait time until the start of the maneuver.

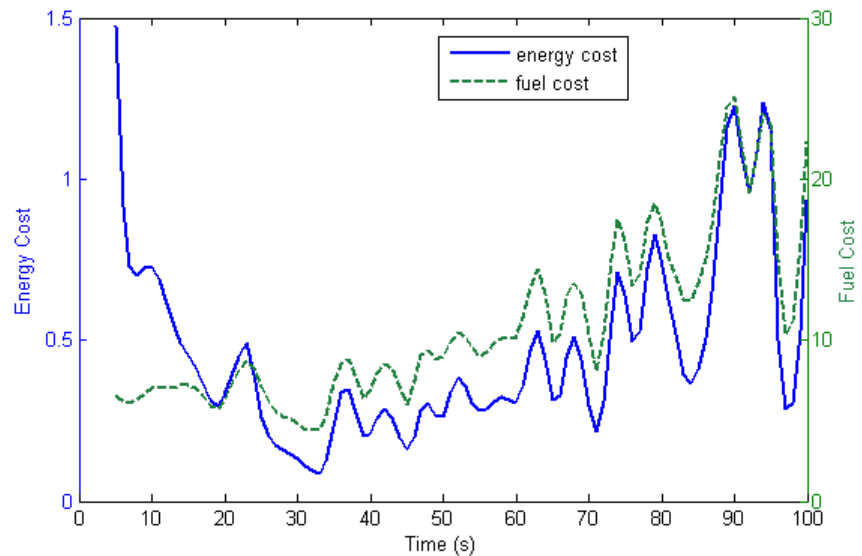


Figure 139. Energy and fuel costs for rendezvous with a tumbling RSO with an asymmetric inertia matrix. The time shown is the total maneuver time.

C. FLEXIBILITY AND ROBUSTNESS OF IDVD TECHNIQUE

The purpose of this section is not to generate an exhaustive list of potential advantages and implementations of the IDVD methods described in the previous section, but to further educate the reader on concepts in order to facilitate new ideas. The minimum energy example was selected for further analysis and implementation based on its nature to minimize control effort over the entire maneuver. It has inherent fuel-minimizing qualities as well as other attractive attributes. The minimum energy cost based maneuver does not have the trait of commanding maximum (or saturated) controls during the terminal phase of docking. The extra control margin makes it the most responsive and safe if sudden changes in trajectory are needed, a potential necessity if closed loop implementation is to be considered. Having low control action also lowers potential effects of plume impingement as opposed to maximum thrust. These combined attributes make it the preferred choice for implementation.

1. Real Time Trajectory Reshaping

Because the IDVD solution is analytic in nature and incorporates prescription of the endpoint values as well as their derivatives, it allows for the possibility of implementation in a closed-loop fashion. If, for example, the minimum energy trajectory (which does not saturate the actuators at any point) solution was calculated using the IDVD method, the resulting varied parameters would vary the higher level derivatives at the endpoints, since the state values and their respective first derivatives were set based on the projected state of the RSO at the final time as well as the initial conditions of the chaser craft. Even after a solution is calculated, if updated information about the current states of the RSO were obtained and the final state values of the RSO were to change (due to unmodeled dynamics or disturbances), this information can be reintroduced into the trajectory generation method to ensure that the endpoints of the trajectory met the desired conditions. Mathematically speaking, the varied parameters, already solved for, would not change, but, keeping the conditions at the beginning of the trajectory fixed, the specified conditions at the ending point, t_f , would be tweaked in order to match the best projected conditions of the RSO docking point, resulting in new coefficients for the

polynomials and therefore instantly reshaping the trajectory. The new trajectory may not be optimized with respect to the slight changes in state variables, but it would be the best solution based on the given constraints (having a feasible trajectory that ends with the desired conditions on the chaser and its docking point) without going through another iteration process. This is in contrast to current iterative real time closed loop optimal-control studies that are performed in simulation, (McFarland et al. 2009), and assume solutions can be obtained in whatever specified update rate is needed, stopping the simulation to compute the optimal control, which can take several minutes to hours depending on computer performance and desired solution resolution. The following two examples are used to highlight the noniterative reshaping trait of the proposed IDVD controller. In the first scenario, the inertia matrix of the RSO is thought to be known as $\mathbf{I} = \text{diag}([3,1,2])$; therefore, it is used to project the states of the RSO at t_f based on the initial conditions of the RSO and Euler's equations for rotational motion. For this exercise, we assume the solution of the varied parameters that optimize the rendezvous maneuver is already known and the conditions are such that the problem is the same as discussed in previous sections based on Table 1. The only difference is that now the actual inertia of the RSO is $\mathbf{I} = \text{diag}([1,2,3])$. The second scenario involves correct knowledge of the RSO inertia, but the presence of an unknown constant thrust in the body frame (such as a thruster that is failed on). New (ideal) RSO state information is available at a rate of 10 Hz, allowing the trajectory generator to reshape in real time based on the new projected endpoint values. A block diagram of the process is shown in Figure 140.

In the previous sections, the IDVD Solver coupled with the Trajectory Generator was treated as the same system. It should be noted that, in fact, they can be separated in order to exploit specific advantages to the IDVD method. The key difference between the IDVD Solver and the Trajectory Generator is the IDVD Solver iterates on the varied parameters (the higher level derivatives of a given trajectory at the endpoints) and uses the Trajectory Generator to solve for the polynomial coefficients and generate the spatial and time trajectories of the system states (as well as the controls). From there, a PI can be associated with the recently calculated trajectory and the IDVD Solver can iterate on

those varied parameters to reduce that PI. In summary, the entire IDVD makes calls to the Trajectory Generator. The Trajectory Generator, when examined alone, takes the endpoint conditions (position and velocity) and higher order derivatives (eg. acceleration) for attitude and translation and solves for the polynomial coefficients using Equations (121)–(128) for attitude and Equations (136)–(139) for translation. The idea being that once the IDVD Solver has values for the varied parameters, the most current information on the RSO docking point can be used to reshape the trajectory, ensuring endpoint conditions are met. Furthermore, it can be implemented in such a manner that also provides the controls necessary to track this trajectory, using the inner-loop controller to take care of small errors (Yakimenko et al. 2008).

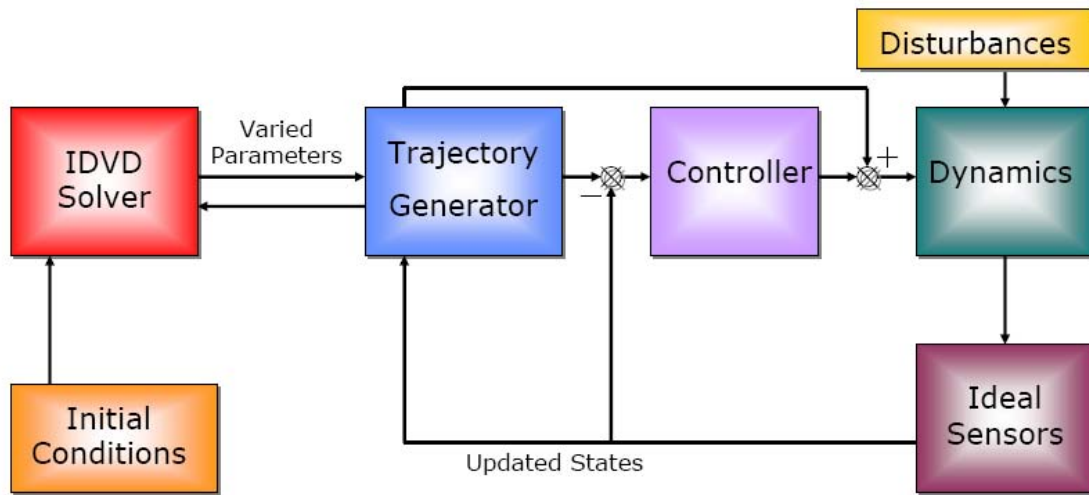


Figure 140. Block diagram of real time trajectory reshaping concepts using IDVD.

a. Solution Results with Inertia Uncertainty

The trajectory resulting from the closed-loop implementation is shown in Figure 142. For this case, the trajectory generator has incorrect information about the inertia of the RSO, specifically it is using a diag ([3,1,2]) inertia matrix (the same as previous examples) when the actual inertia matrix is diag ([1,2,3]). This is a more extreme case of misidentification and was chosen to best illustrate the concept. In actuality, the inertia matrix information of the RSO employed by the trajectory generator would most likely be more accurate, or simply use an identity inertia matrix.

In Figure 141, the actual trajectory of the RSO docking point is illustrated by a dotted red line with a circle marker highlighting a subset of points at which updates are provided to the RSO. The dash/dot lines illustrate some of the projected trajectories of the RSO docking point based on the most current angular position and velocity information, propagated with the (incorrect) RSO inertia matrix currently used by the trajectory generator. The dotted line shows the projected endpoint of the RSO docking point as it progresses through each iteration. The projected RSO docking point and the actual RSO docking point converge to the same value as at t_f , since the amount of time for the trajectory generator to project into the future, based on incorrect inertia information, where the RSO docking point will be decreases. The thin colored lines show a subset of the reshaped trajectories at each timestep. The blue dots show the resulting overall trajectory for the chaser vehicle, overlaid with intermittent models showing the resulting attitude of the vehicle.

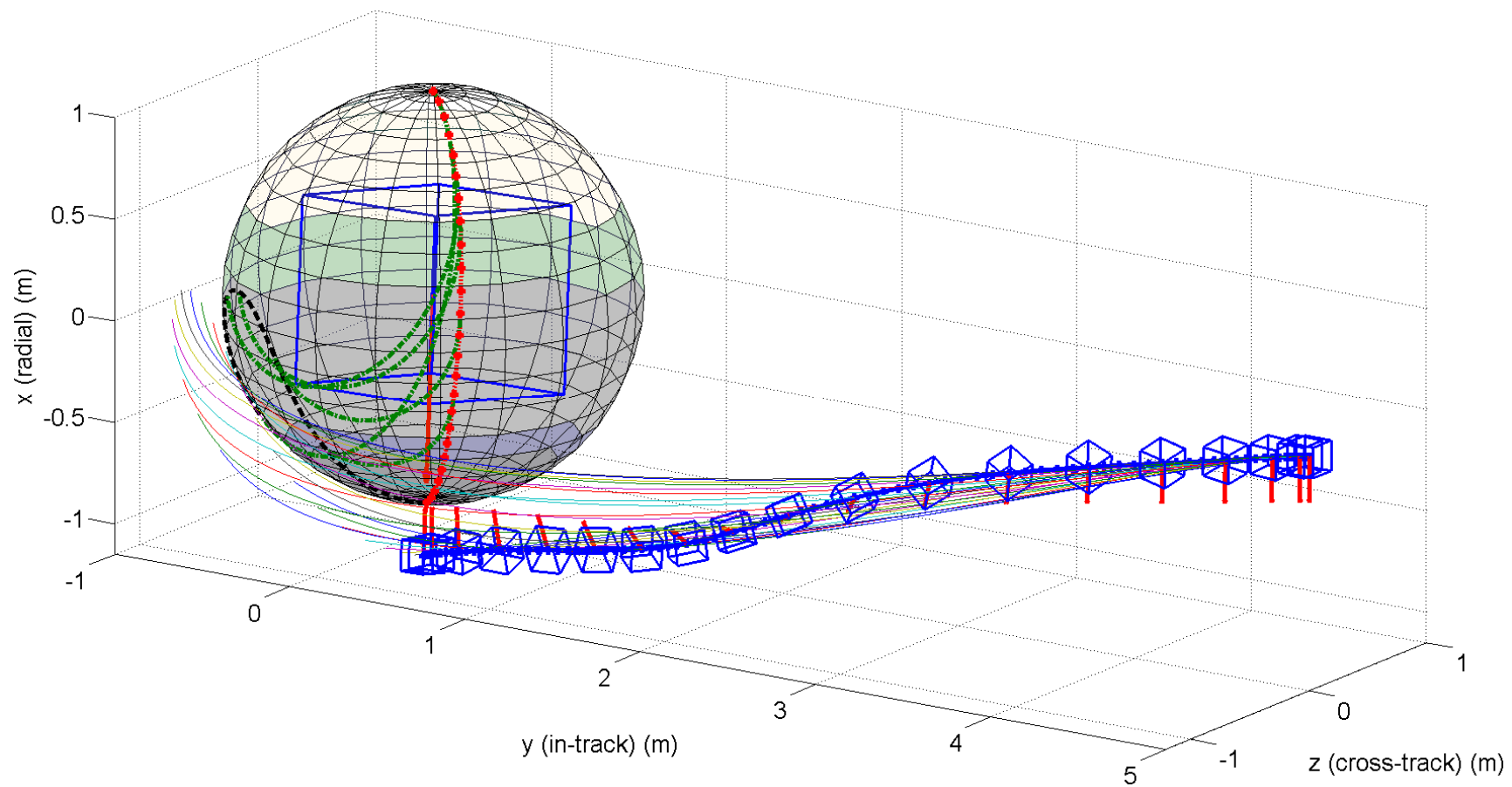


Figure 141. Results of the final trajectory for the close approach example having a misidentified inertia matrix IDVD real time reshaping.

Figure 142 shows a close-up, expanded view of the previously described trajectories. Information shown is the actual trajectory (red, circle), the projected endpoint (black, dash) and a sample of the overall projected trajectories based on state updates (green, dash/dot). Figure 143 illustrates the evolution of the current trajectory over time and the position of the chaser CM.

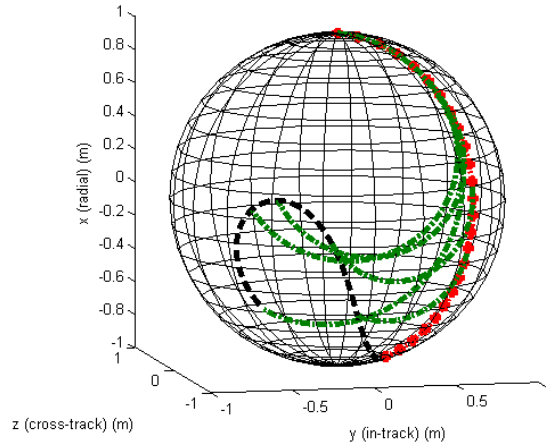


Figure 142. Exploded view of the RSO docking point information for a misidentified inertia matrix example.

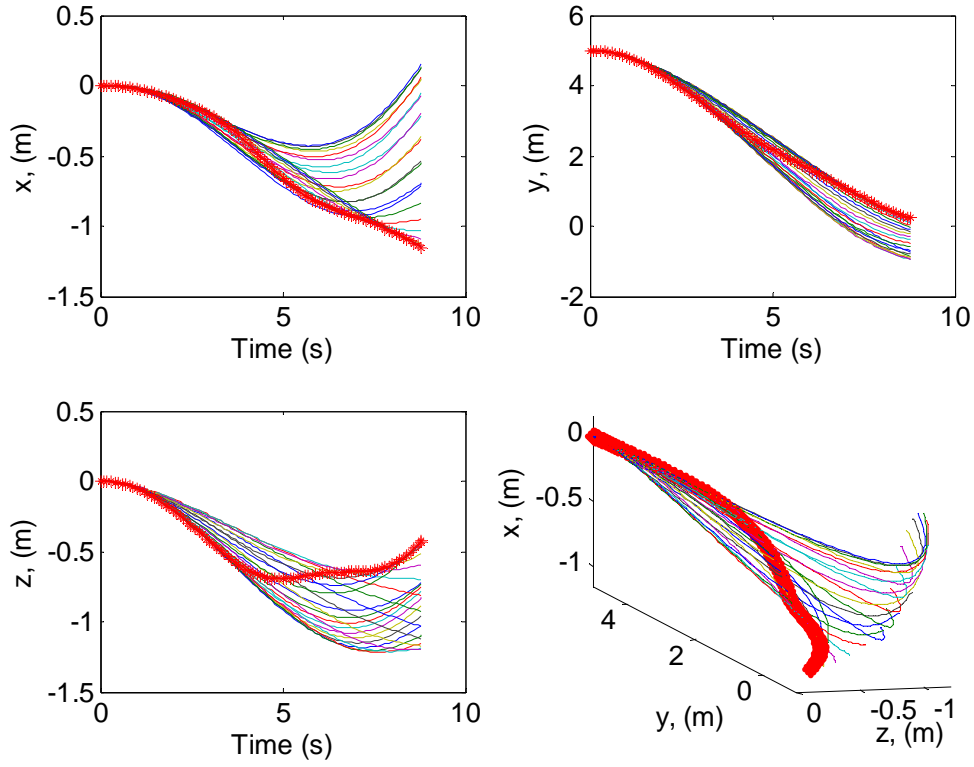


Figure 143. Spatial views of the resulting trajectories for the close approach example having a misidentified inertia matrix IDVD real time reshaping.

b. Solution Results with Unknown Constant Torque

For the next example, it is assumed the inertia matrix of the RSO is known by the trajectory generator, but an unknown torque exists, acting on the vehicle in the x-body frame. This can be thought of as a thruster that is stuck in the On mode, thus generating the disturbance. The resulting plots, similar to those in the previous section are shown in Figures 144 and 145. Figure 144 shows several of the reshaped trajectories, along with the overall trajectory for the chaser vehicle, while Figure 145 shows the time history for each translational component of the Chaser vehicle, as well as the evolution of the reshaped trajectories. As in the previous example, the projected and actual endpoint of the RSO states converge to zero as the chaser vehicles converges to the final state.

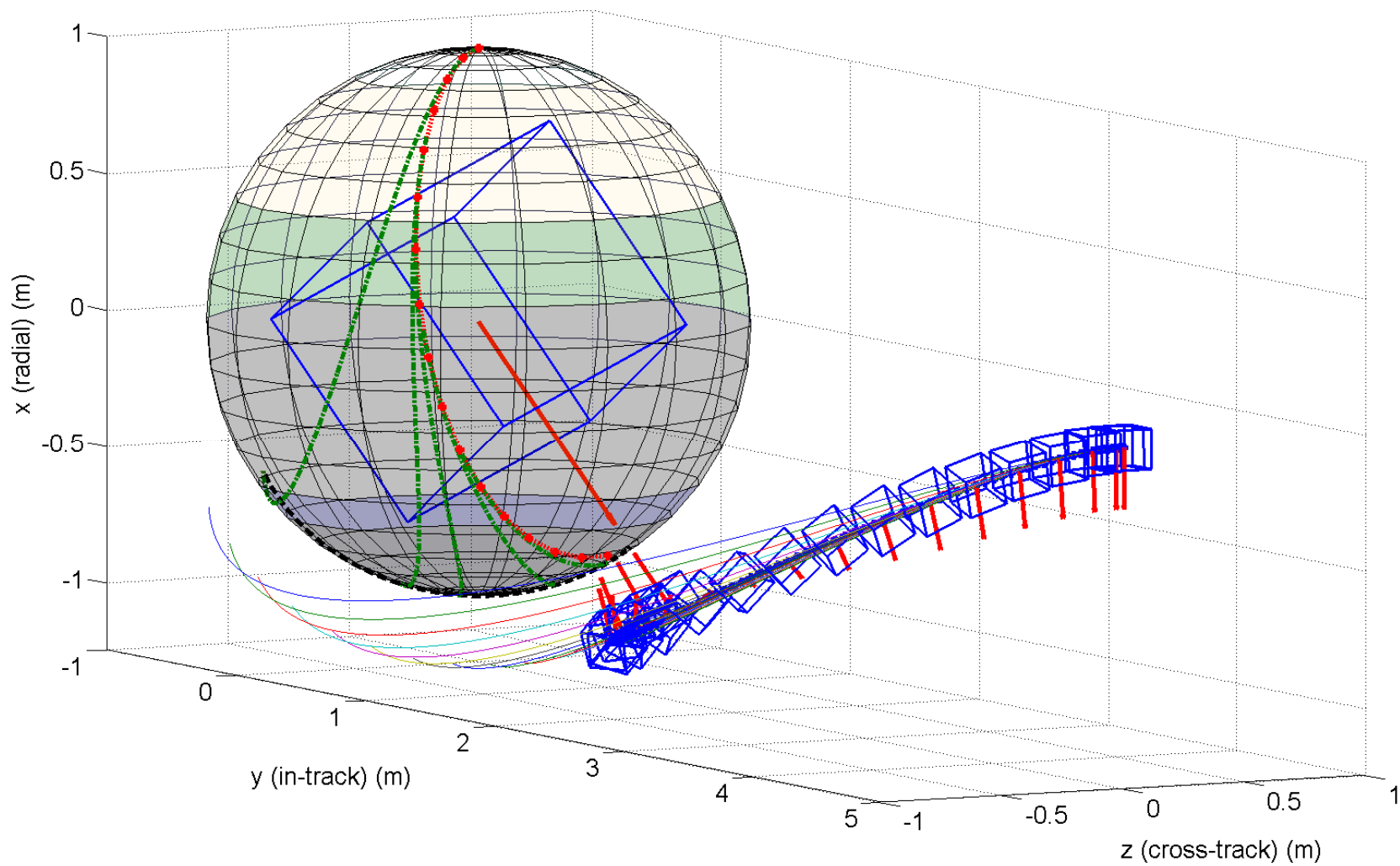


Figure 144. View of the close approach trajectory example with an unidentified, constant torque using IDVD real time reshaping.

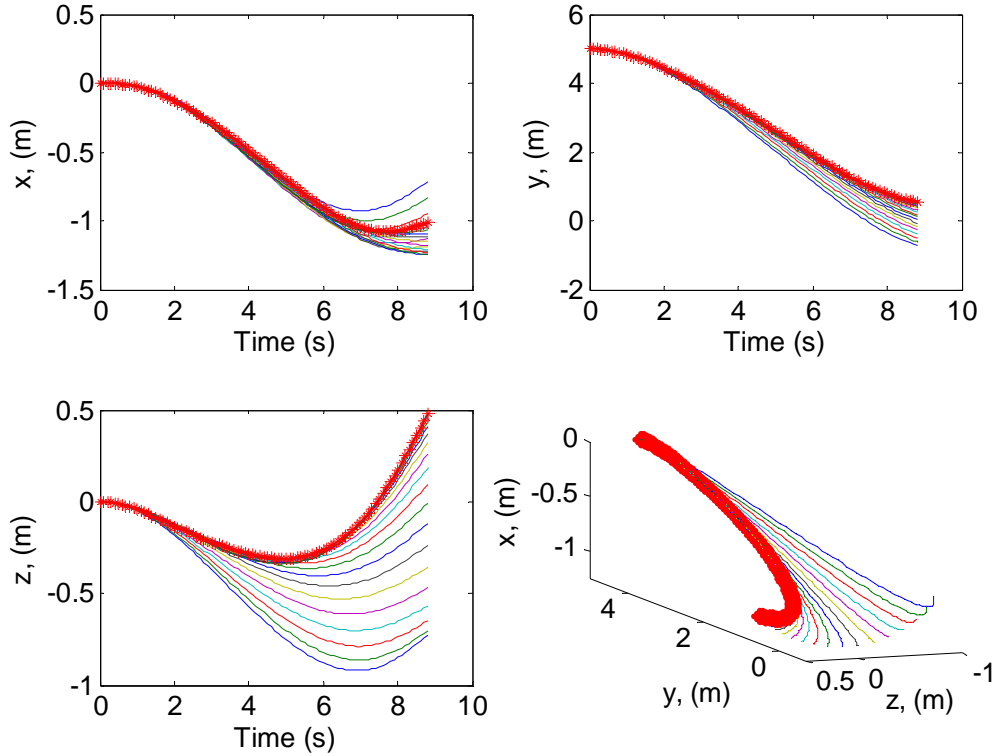


Figure 145. Spatial views of the resulting trajectories for the close approach example having an unidentified constant torque using IDVD real time reshaping.

2. Resolving and Reoptimizing the Problem in Real Time

If the solution of the IDVD could be computed in real time, with the definition of real time based on the requirements of the user/customer and the estimation attributes of the sensor system, the problem can be recomputed as the trajectory is traversed. An illustrative example of this concept is given in Figure 146. In this case, a switch is installed that allows the initial conditions to be updated based on the most current system states. The problem can then be resolved using IDVD and the new resulting trajectory immediately updated. The key difference between this implementation and the reshaping approach is that the trajectory is resolved and reoptimized based on the most current states of the RSO and the chaser vehicle. Even though for this case the trajectory is resolved, one of the greatest benefits between the IDVD method is still that if the endpoint conditions change, the IDVD can use this information to tweak the trajectory.

Other methods, specifically pseudospectral, must resolve the entire problem in order to obtain a trajectory that, when the controls are propagated, would finish the maneuver in the correct position with the desired conditions.

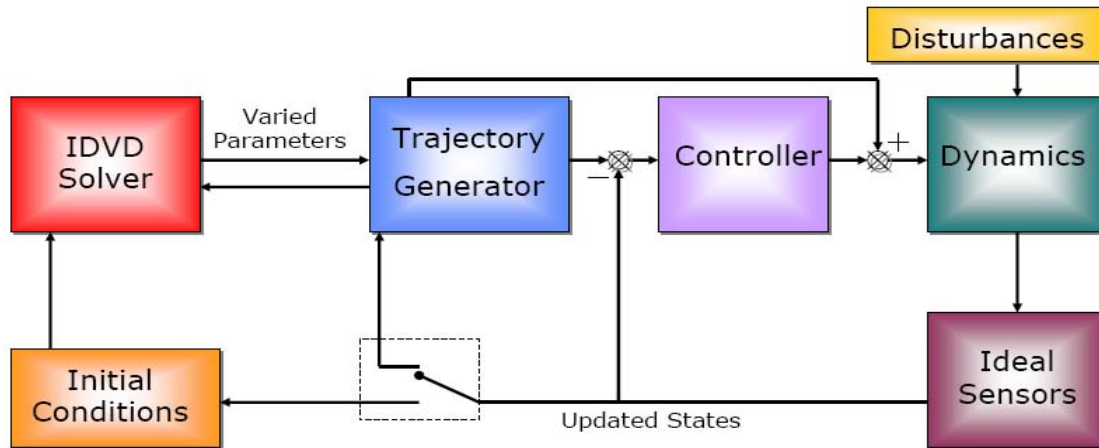


Figure 146. Block diagram of real time trajectory recalculation concepts using IDVD.

This concept is demonstrated on the same example stated in Chapter VII.D.1.b, employing the inertia uncertainty. The trajectory generator is hampered by not knowing the true inertia of the RSO, but is receiving state updates and recalculating the trajectory at a rate of 1Hz. This allows the current trajectory to be based on the most current known states of the chaser and the RSO. The switch allows for the trajectory generation method to use the rapid-resaping concept described in Chapter VII.D.1 with a 10 Hz update rate for the final 1 second, or endgame phase, of the trajectory since uncertainties in guidance may need to be corrected much more rapidly (Vaidyanathan et al. 2001). This also demonstrates the IDVD's ability to be integrated with other guidance systems to construct the best overall system.

Figure 147 shows the plots of each recalculation (top), as well as the overall resulting trajectory (Bottom). The multicolored segmented lines in the top figure illustrate each trajectory recalculation, occurring at 1-second intervals shown by the blue dots. The segmented line on the 1m sphere shows black "x" marks representing the current projected location of the docking point at the calculated t_f for each optimized trajectory.

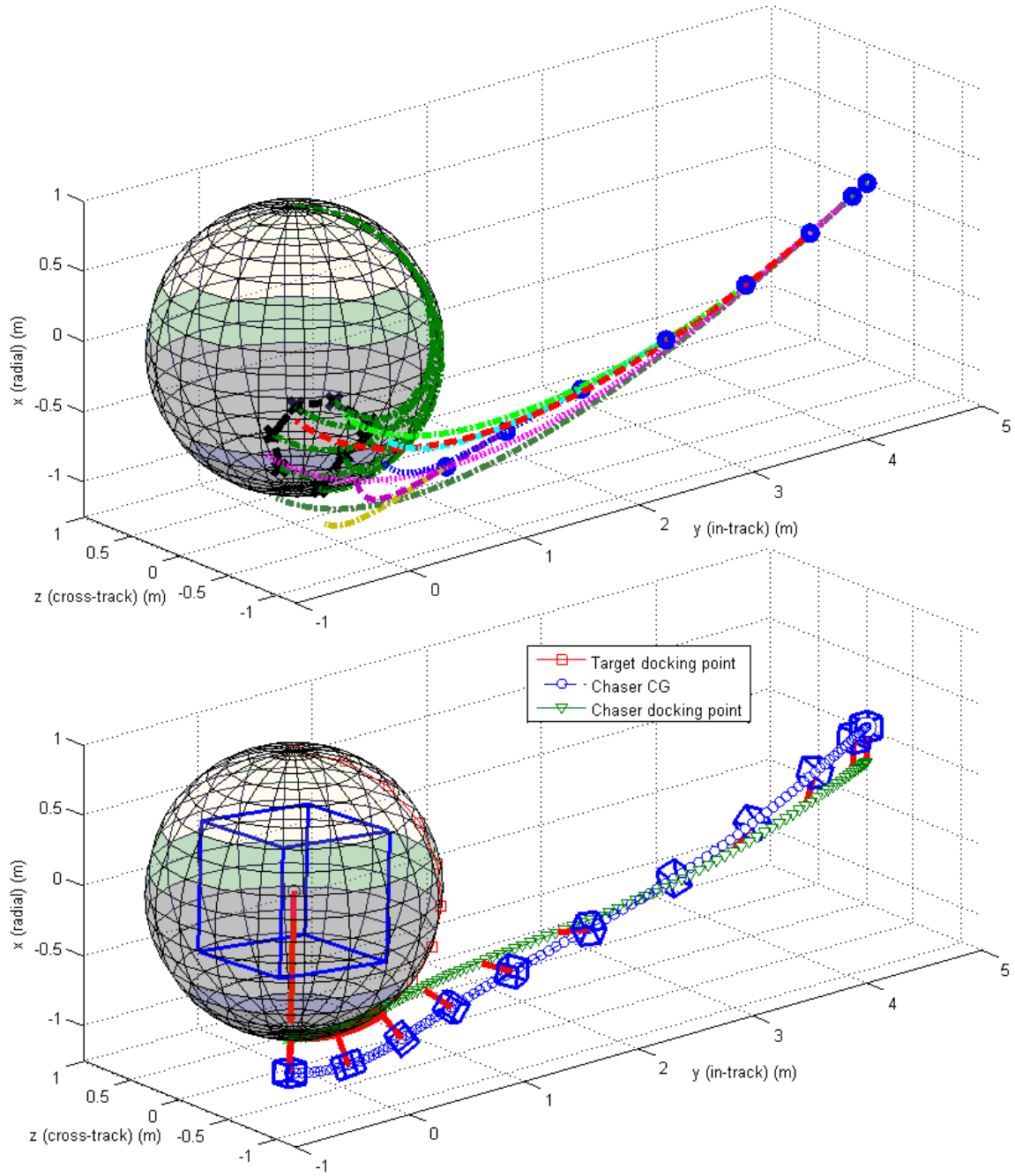


Figure 147. (Top) Overlaid recomputed trajectories. (Bottom) Resulting overall trajectory based on the most current recalculation.

3. Specific Waypoint Spacing in the Time Domain

Another advantage to the specific IDVD formulation, and an enabling attribute that allowed the implementation of the closed-loop implementation concepts from the

previous sections, is the freedom to choose node or waypoints. In previous work (Bevilacqua, Romano and Yakimenko 2009), integrating the inverse of speed factor that was based on switching times led to the user to be at the mercy of node spacing that is dependent upon the history of the speed factor itself. This could lead to gaps in control and trajectory information that make the solution incapable of implementation. This is also seen in application of pseudospectral direct methods (Hurni 2009) while using a commercial program for direct optimization of trajectories and having to add layer of control (and complexity) in order to correctly interpolate control information in between the node points of the solution. Because of this, for the problem studied concerning a wheeled rover-like vehicle, the user is forced to use velocity and heading as a control vector, a layer above the intuitive controls of acceleration and angular rate. An example where the constraints on the previous problem are relaxed to illustrate how the nodes of the solution get bunched together in regions where the speed factor is the greatest is considered. This is because the node spacing is uniform with respect to the virtual argument τ , as shown in the bottom plot of Figure 148.

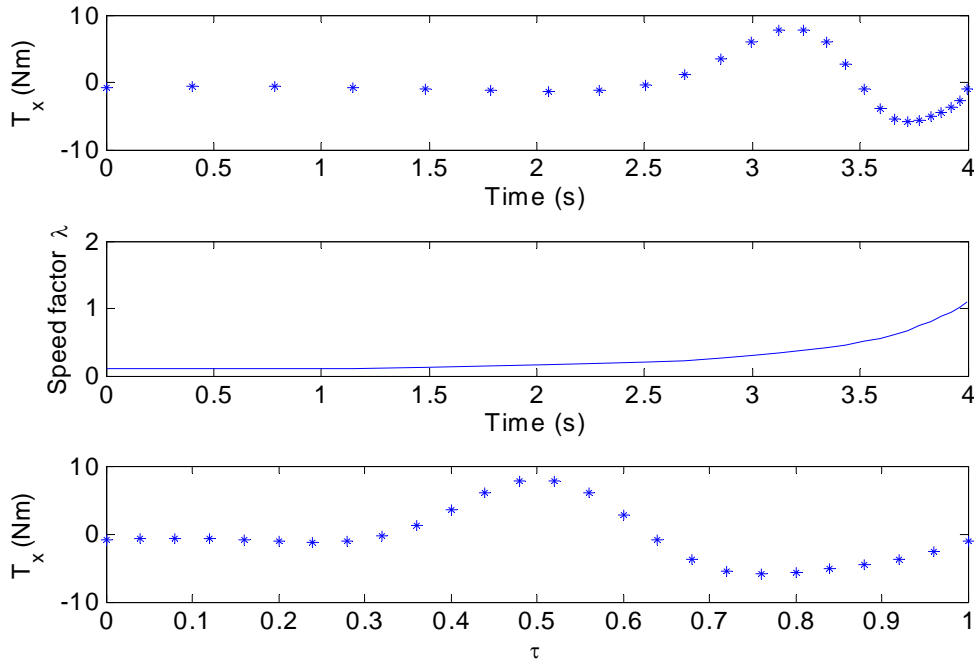


Figure 148. Control history of a representative trajectory with uniform node spacing in the virtual domain.

The question posed is what dispersion of node points in the virtual domain will guarantee uniform node spacing in the time domain. For the methods described in Chapters III and IV, the speed factor was chosen such it has an analytic expression for the integral of its inverse, which maps a specific value of t to the time domain as shown below:

$$t = \int_0^\tau \frac{1}{\lambda(\zeta)} d\zeta. \quad (149)$$

This results in the ability to express τ as a function of t and vice versa.

Once the final time is obtained from the initial solution, a set of evenly spaced nodes (or arbitrarily spaced nodes based on any inputs from the user) set in the time domain can be instantaneously converted to the virtual domain. This new set of nodes in the virtual domain, when implemented, leads to the evenly spaced nodes in the time domain as shown in Figure 149.

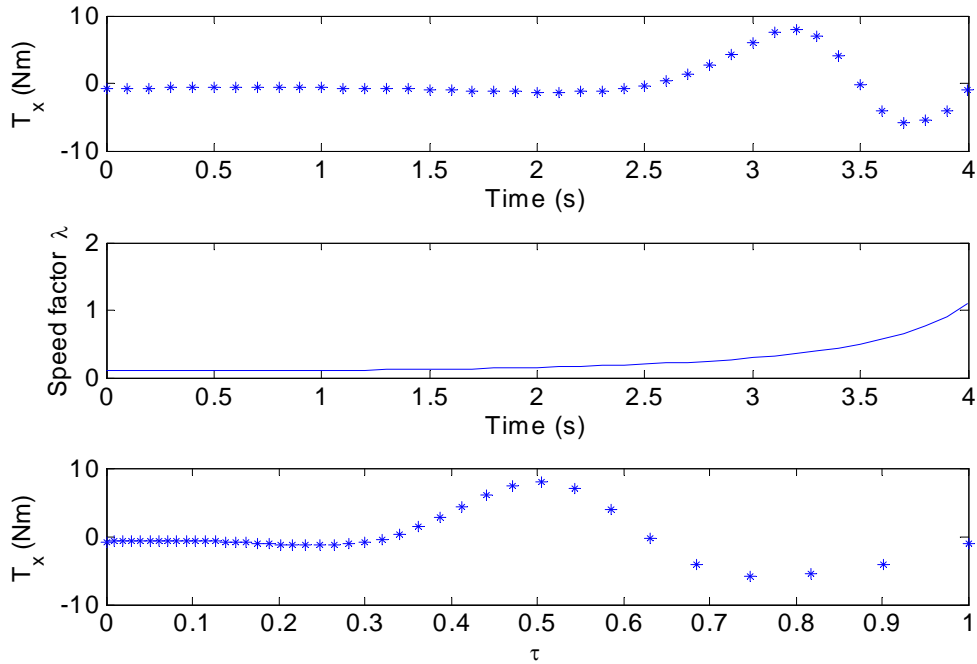


Figure 149. Control history of a representative trajectory with uniform node spacing in the time domain.

D. EXTENDED EXAMPLES USING INVERSE DYNAMICS IN THE VIRTUAL DOMAIN

Up to this point, the same initial conditions have been used to analyze the optimal solution to a close approach problem, compare with a suboptimal rapid-trajectory generation technique, and demonstrate potential real time close-loop applications. The IDVD (3rd) and IDVD (3rd–4th) method generated in the previous chapter is applied to several pseudorandom initial chaser conditions and scenarios. First, 10 samples were developed to apply the IDVD technique. From these 10 samples, the initial velocities of the spacecraft were assumed to be zero and the initial orientation of the chaser is assumed to be coincident with the orbit frame; both assumptions were taken from initial conditions used by McCamish (2007). The pseudorandom initial positions of the chaser are chosen by first determining a vector then multiplying it by a uniformly distributed random number between 5 m and 10 m as shown in Equation (150).

$$\begin{aligned} x &= \sqrt{(1 - rand_x^2)} \cos(rand_\theta) rand_r \\ y &= \sqrt{(1 - rand_x^2)} \sin(rand_\theta) rand_r \\ z &= rand_x rand_r \end{aligned} \tag{150}$$

with $rand_x \in [0 \ 1]$, $rand_\theta \in [0 \ 2\pi)$, $rand_r \in [5 \ 10]$ having uniform random distributions. The desire was to have an equally distributed set of points in range (from 5–10 m) and all directions, understanding that based on the given formula, the points would not necessarily be uniformly distributed about the volume housing those points. The solutions to the first ten generated initial conditions using IDVD (3rd) are illustrated in Figure 150. Again, the initial conditions of the RSO were kept consistent with the previous examples from this manuscript, having a $\mathbf{I} = \text{diag}[3 \ 1 \ 2]$ and initial angular velocity of 0.25 radians/second in the y and z body frame and initial $\mathbf{q} = [0 \ 0 \ 0 \ 1]$. For the first set of ten shown in Figure 150, the average computational time was 21.0 seconds.

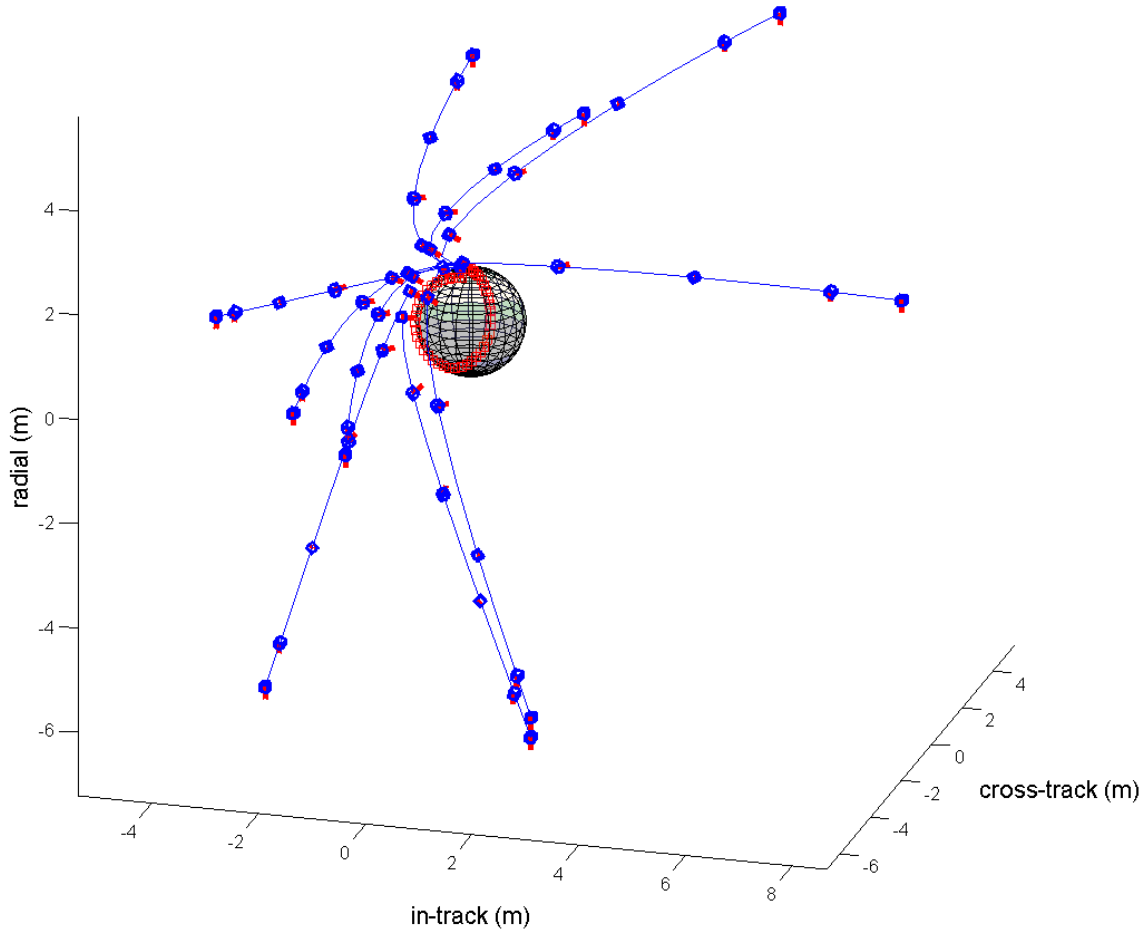


Figure 150. Trajectories for sample rendezvous problem for the first 10 random starting chaser positions with zero initial relative velocity.

Next, over 1000 pseudorandom initial conditions were generated with varying initial relative velocity as well as position. The maximum magnitude of the initial velocity was limited to 1 m/s to be consistent with “relative speed limit” constraints placed during close proximity operation again in the complementary work of McCamish (2007) as well as having the body frame aligned with the orbital frame. This would be consistent with a GNC concept utilizing McCamish’s Artificial Potential Function modified algorithm up until it is time to perform the close approach for docking maneuver when the RSO has nonzero angular rates. The solutions to the first 10 samples are shown in Figure 151. Notice the presence of initial velocity for the chaser vehicles results in a modified path to the final conditions. This is due to the fact the resulting

polynomial trajectory must shape itself to have a directional component at the beginning of the maneuver that will accommodate the different initial conditions.

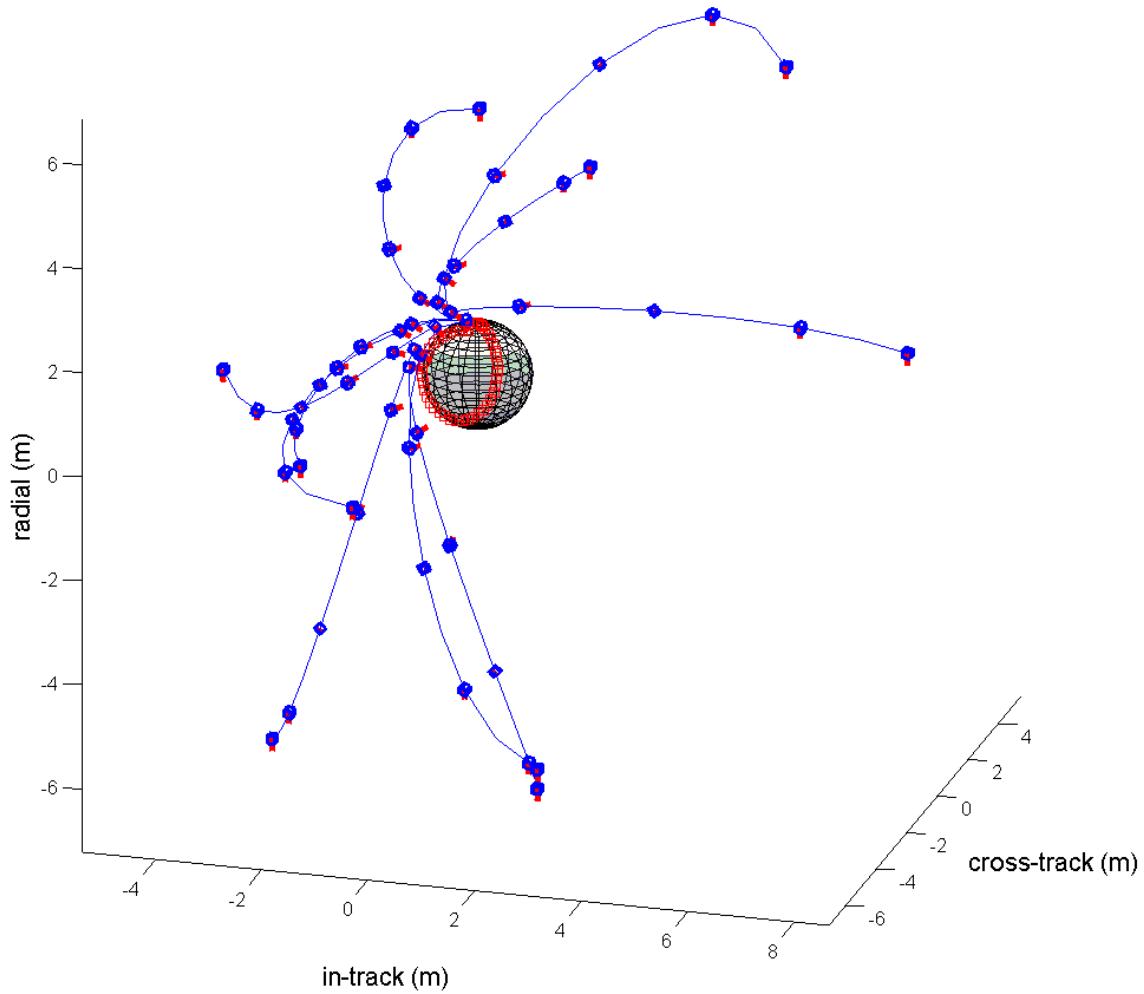


Figure 151. Trajectories for sample rendezvous problem for the first 10 random starting chaser positions with nonzero initial relative velocity.

The complete distribution of initial conditions is shown in Figures 152–155. The maximum allowable final time was also increased to 15 seconds for the ensuing calculations.

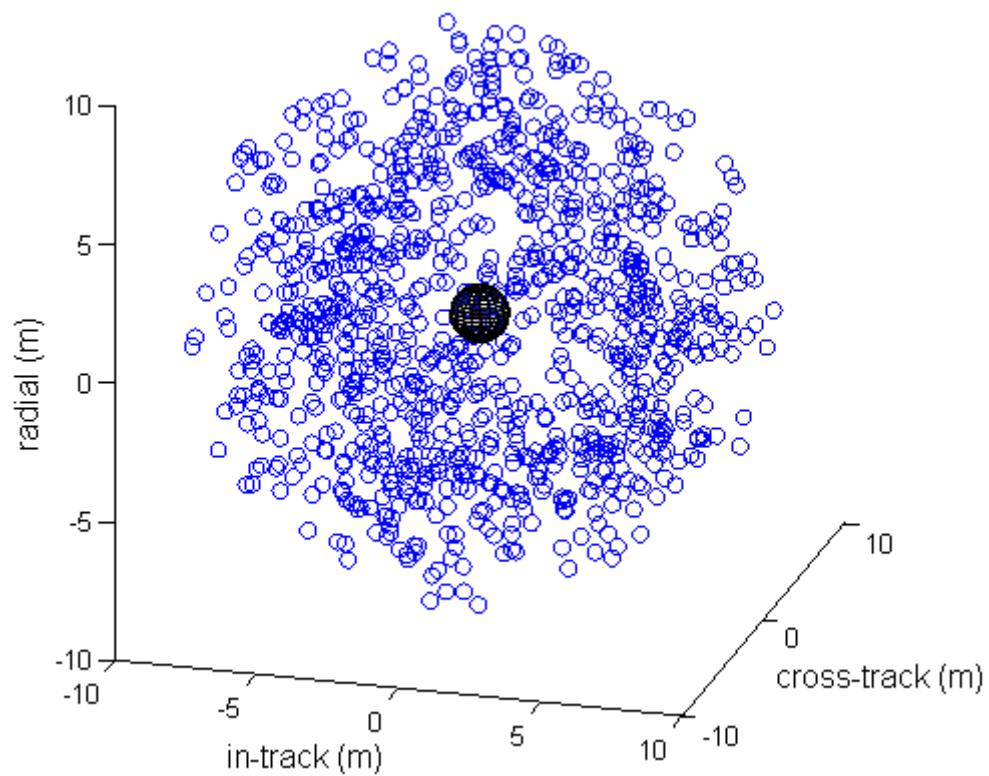


Figure 152. Distribution of the 1000 pseudorandom initial conditions tested.

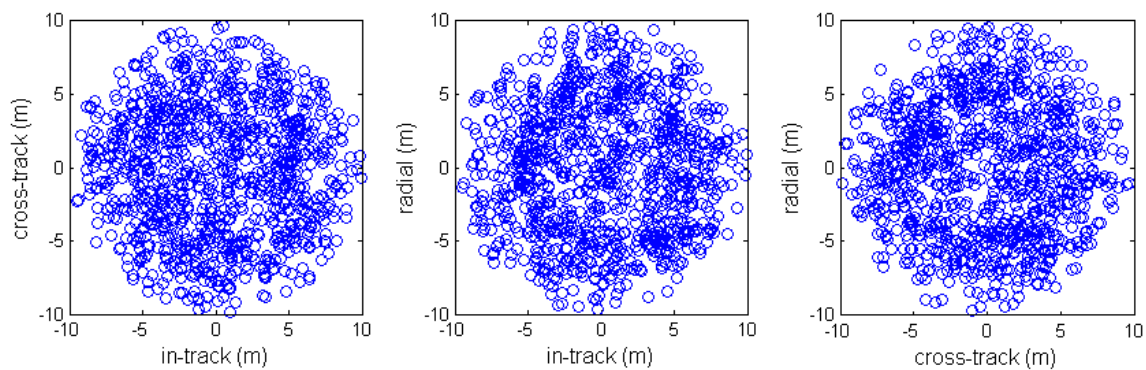


Figure 153. Planar view of the 1000 pseudorandom initial conditions tested.

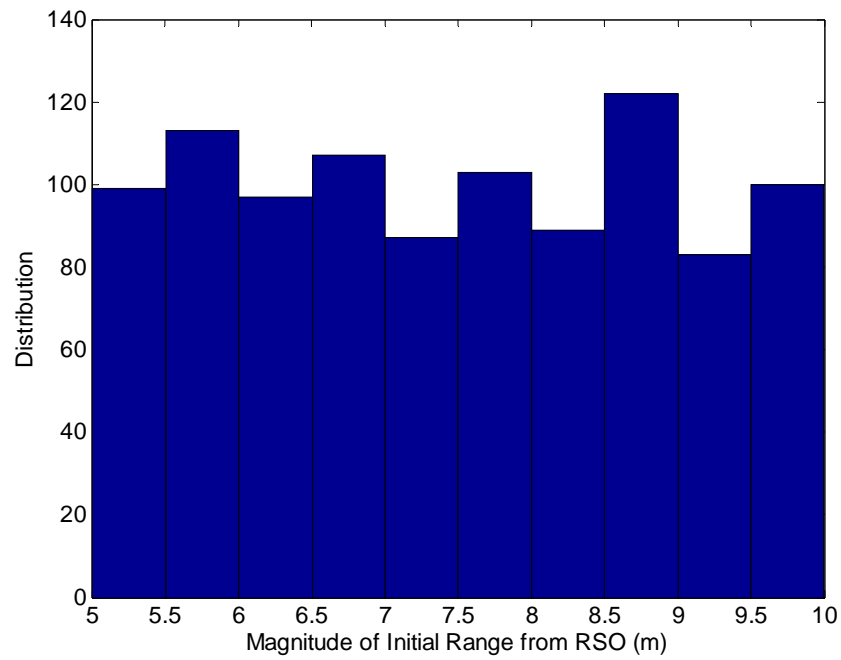


Figure 154. Distribution of the initial chaser range for the 1000 pseudorandom samples generated.

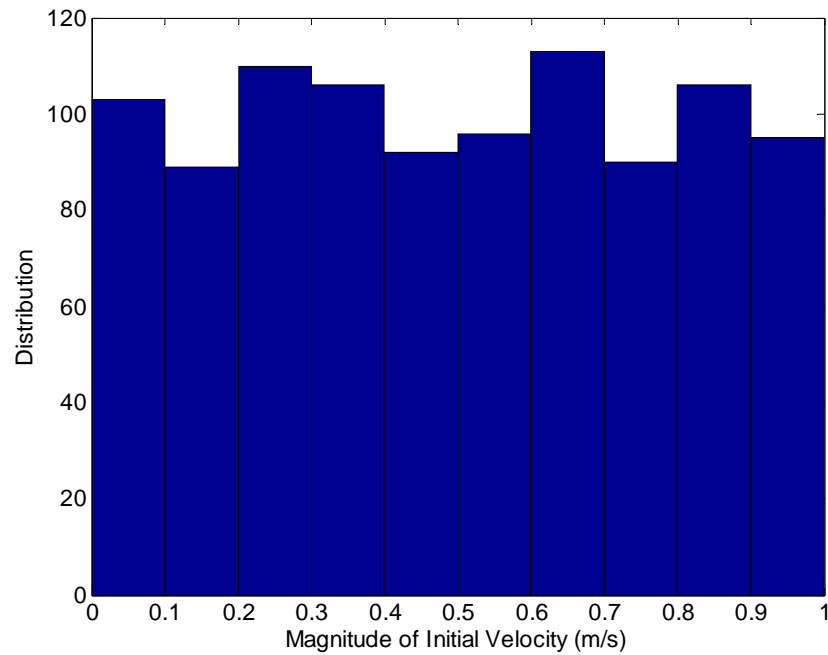


Figure 155. Distribution of the initial chaser velocity for the 1000 pseudorandom samples generated.

The IDVD (3rd) method was run using the 1000 sample and the mean time to calculate the solution was 34.3 seconds. The mean time to complete the maneuver was 14.3081 seconds and the mean energy PI was 0.2289. The same 1000 initial condition set was also run using the IDVD (3rd–4th) method. The mean of the energy PI was 0.2112, but the average run time was 228.9097 seconds. A summary of the results is shown in Table 28 for initial conditions on relative position and velocity pertaining to Figures 154 and 155 and Table 29 pertaining to initial relative positions based on Figure 154 but starting with zero initial relative velocity. Table 30 shows the results for varying the initial conditions on position and velocity of the chaser vehicle according to Figures 154 and 155 but also varying the initial angular velocity of the RSO based on Figure 156. Although the IDVD (3rd–4th) method results in an overall average cost over the sample of initial conditions tested, the computational time was substantially more.

	IDVD (3rd)	IDVD (3rd-4th)
mean values	Min Energy	Min Energy
Final Time	14.3081	14.5191
Energy	0.2289	0.2112
Fuel	4.3124	4.2516
CPU Time (sec)	34.3	228.9

Table 28. Summary of performance indices and computational time for IDVD (3rd) and IDVD (3rd–4th) method using 1000 pseudorandom samples for varying the initial conditions of the chaser position and velocity.

	IDVD (3rd)	IDVD (3rd-4th)
mean values	Min Energy	Min Energy
Final Time	14.3094	14.6107
Energy	0.1742	0.1631
Fuel	3.8773	3.8544
CPU Time (sec)	32.9	191.0

Table 29. Summary of performance indices and computational time for IDVD (3rd) and IDVD (3rd–4th) method using 1000 pseudorandom samples for varying the initial conditions of the chaser position but having the initial relative velocity be equal to zero.

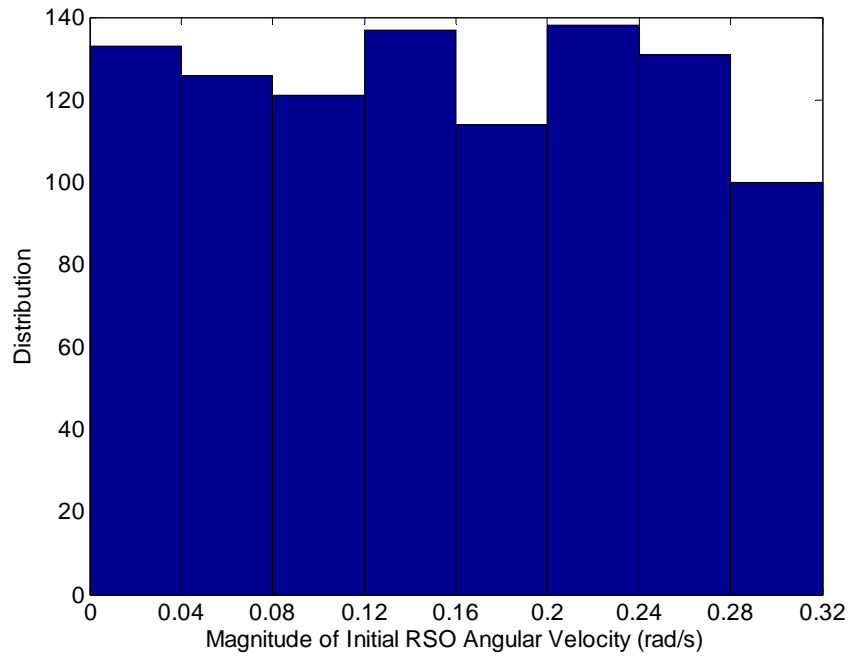


Figure 156. Distribution of the initial RSO angular velocity for the 1000 pseudorandom samples generated.

	IDVD (3rd)	IDVD (3rd-4th)
mean values	Min Energy	Min Energy
Final Time	14.9716	14.9799
Energy	0.2138	0.1689854
Fuel	3.3867	3.2472
CPU Time (sec)	20.0	228.5

Table 30. Summary of performance indices and computational time for IDVD (3rd) and IDVD (3rd–4th) method using 1000 pseudorandom samples for varying the initial conditions for position and velocity of the chaser and initial angular velocity of the RSO.

VII. CONCLUSION

This section summarizes the contributions and findings of this dissertation and is organized as follows: First, a summary of the research is presented. This is followed by a summary of the contributions of this research followed by a discussion of the advantages and disadvantages pertaining to the different methods. Finally, a recommendation for future study is presented.

A. SUMMARY OF RESEARCH

A six DoF, 20-state model of two spacecraft rendezvous is developed, one of which was controlled, the other considered to be passively tumbling. A solution is obtained for the problem of close approach, up to the point of contact, using a direct optimal control method. The solution is then verified as optimal by way of an indirect method based on the MP. Next, a trajectory generation method for spacecraft reorientation is developed, based on a quaternion construction of IDVD. This new construction enables implementation of an IDVD trajectory generation method for the problem of a spacecraft performing a close approach maneuver to a tumbling object. Finally, the advantages of the IDVD method were exploited and demonstrated in simulated scenarios that employ closed-loop feedback.

B. SUMMARY OF CONTRIBUTIONS

Contributions of this dissertation are considered to have two distinct categories of impact, theoretical contributions, enhancing the overall understanding contribute to the overall body of knowledge, and practical contributions, advancing the current state of the art and moving the technology closer to implementation. They are both summarized in the next several paragraphs.

Summarizing, the minimum quadratic-control, minimum fuel and minimum time continuous optimal control 3D spacecraft rendezvous-docking problems are formulated for the first time in literature, and addressed using a direct collocation method, GPOPS. For both problems, the desired optimal trajectory of chaser spacecraft with respect to a

tumbling RSO is sought such that the desired docking points match in position and velocity. Moreover, the solutions obtained were verified in the numerical simulations based on the Minimum Principle. Those solutions included derivation of the adjoint equations, formulation of the necessary conditions for the optimal solution, and synthesis of the optimal controls. The results obtained from using the direct collocation method—in this case the open source software of GPOPS—are very close to those obtained using with the Minimum Principle. It was also found that path constraints are necessary when solving for the optimal trajectory in order to prevent undesired collision of spacecraft. This was shown by the active path constraint that results in discontinuous costates upon contact with the constraint boundary.

As expected, the GPOPS direct collocation (pseudospectral) method was found to be quite reliable and yielded the results relatively fast. Moreover, these results were also validated via propagation of the states based on the system dynamics, and optimal controls were found. That is what would be needed for possible onboard implementation of the developed algorithms. However, the GPOPS direct collocation method proved to be unable to produce the results fast enough so that it could be used in real time. Hence, in the nearest future, it only leaves off-line open-loop option to be possibly used on the real spacecraft. But even then, it was found that in case of a singular control, the method returns somewhat infeasible results that cannot be implemented. The main simplifying hypothesis, considered in the numerical simulation presented in this section of the research, was the spherical inertial symmetry of both chaser and target spacecraft, as well as thrust control limits being applied in the orbital frame regardless of chaser spacecraft orientation. In order to achieve feasible controls in real time, this assumption needs to be removed, along with investigating the usage of different combined PIs and other direct methods based on inverting of the dynamics of the problem.

This research then continued work on spacecraft rendezvous by modifying the previous problem formulation to include body mounted translational thrusters on the chaser spacecraft, as well as extensions on the final conditions. These new final conditions involved matching the velocity and position of the spacecraft docking points, as well as matching angular velocity and orientation. Inertia parameters on the RSO were

also set to different values, removing the assumption of an identity inertia matrix, in order to stimulate dynamic angular rates in all three axes. The problem was addressed again using the most plausible pseudospectral methods. In addition, the optimal control structure was synthesized and analytical expressions for the transversality conditions were derived (exploiting the Minimum Principle) to compare those with the calculated solution. In this section of research, three PIs were examined. It was found that the minimum time solution exhibits the expected bang-bang control and the minimum fuel solution exhibits bang-off-bang characteristics. Assuming controls in the body frame as opposed to the orbital frame (as in the previous work by the authors), while making the problem more complex, seems to alleviate the singular control that appeared in the z-translation, and therefore, made the GPOPS solution feasible. Another finding was that instead of addressing the minimum fuel problem, it was wiser to consider a minimum-energy problem instead. The reason for this is that although the minimum fuel solution happened to be about 30% more effective, it would be difficult to implement the control in practice because of the bang-off-bang control structure. The minimum energy solution, which lasts longer (but follows a similar path), assures smooth, easy-to-follow control time histories. Furthermore, the minimum energy solution does not involve having the actuators saturated at the final time (a characteristic of bang-off-bang maneuvers for minimum fuel or bang-bang for minimum time), increasing the safety of the maneuver by having control margin available if a rapid corrective maneuver is needed and reducing the likelihood for or large disturbances due to plume impingement. The minimum time solution (following a different path) does not contact any state constraints and therefore has continuous costates, while the minimum-control and minimum energy solutions contact the keep-out zone of the RSO, while approaching for docking. Finally, it was shown that the required CPU time greatly exceeds that of the maneuver itself, which as of now excludes pseudospectral direct collocation methods from being a candidate for onboard application. Instead, exploitation of other approaches, willing to sacrifice a fraction of the PI but decrease the required CPU time by several orders of magnitude, were investigated, as summarized below.

In the case of reorientation maneuvers, the IDVD method with the novel quaternion approximation functions proposed in this dissertation allows computing feasible solutions fast enough to be used onboard satellites for on-line computation of slew maneuvers. Moreover, because of the smooth controls' histories it can be implemented in the control schemes involving a feed-forward loop. For this specific implementation a form of the speed factor was chosen based on a combination of nonlinear equations as opposed to integrating dynamic equations based on having switching times of the controls act as varied parameters (Yakimenko 2000). Because of this, compared to the true time-optimal solutions, the IDVD trajectories do not have the capability to generate a bang-bang control solution, which results in a slightly worse PI. However, smooth controls benefit other mission preferences of having desired rates at the endpoints as well as guaranteeing the endpoint constraints of the maneuver are always satisfied. In addition, it is a definite advantage in rapidly changing acquisition or tracking scenarios and when the slewing spacecraft possesses low frequency flexible modes. This formulation is later applied to situations where the attitude is coupled with other dynamics such as translational motion in rendezvous and docking applications. In this case, a simple eigenaxis slew would not meet mission criteria such as matching rotational motion.

IDVD is further applied to the maneuver that has both translational and attitude motion, such as the case with the focus of this research involving close approach to a tumbling object. The IDVD method developed is highly flexible, allowing for matching any mixed set of endpoint conditions, generating user specified waypoint spacing, and providing a mechanism for real time closed-loop trajectory reshaping. Based on the trajectory generation results for the benchmark 3D scenario, the minimum-control solution was chosen for further study and implementation. This was due to its inherent characteristic to conserve fuel, while providing specific safety qualities, such as not commanding actuator saturation (which would limit its ability to do rapid reactive maneuvers in case of closed loop implementation of higher-level emergency abort).

Within the IDVD method, there are still variations in trajectory parameterization techniques. The IDVD 3rd method employs varying the 3rd derivative at the endpoints of the trajectory while the IDVD 3rd–4th method uses both the 3rd and 4th derivatives as varied parameters. Because of this, the IDVD 3rd–4th method can provide a final trajectory that is lower in PI but at the cost of computational time. The IDVD 3rd method can obtain a solution that is within 10% of the IDVD 3rd–4th method while only using 15% of the computational time, based on results from 1000 samples with random initial conditions.

C. CONSIDERATIONS AND RECOMMENDATIONS

Trajectory generation by direct collocation methods have the benefit of providing a mathematically verifiable optimal solution. The major drawback is the excessive computational time needed on current computing systems. The resolution of the solution, and computational time, is highly dependent on the amount of nodes used to solve the problem. Furthermore, the user cannot specify where the nodes are placed and this leads to complex interpolation schemes or adding an extra layer of control, increasing the complexity of the problem formulation in order to have sufficient control information to implement. The benefits seen by IDVD methods are the rapid computational time that allows for a feasible solution to be generated, potentially onboard a spacecraft and in closed-loop. Because of the analytic nature of its solution, high node trajectories can be generated without sacrificing computational time. The ability to reshape itself when provided updated information on the RSO states and the docking point reinforce the overall robustness of the IDVD method for trajectory planning. By increasing the order of the basis polynomials, thereby, increasing the varied parameters, solutions can be obtained that better minimize the PI, but at the cost of higher computational times. Furthermore, the node spacing of the control solution can be completely specified by the user, therefore, eliminating the previously discussed error incurred by interpolation of potentially complex control profiles or adding extra layers of control because of the lack of flexibility in the node spacing in the control solution set by the direct collocation methods, which are not guaranteed to be continuous. It should be noted that using the IDVD formulation, constraints can be set on higher derivatives than the level of the

control vector. For example, constraints on the jerk profiles can be enforced, while still using the acceleration profile as the control vector. This attribute, which is in complete contrast other direct methods, is based on the analytical formulation of the trajectory. For example, to set constraints on jerk using direct collocation method techniques, the control vector would have to at least be based on jerk (or even higher derivative) and not acceleration, again adding additional levels of complexity.

Drawbacks to the IDVD method are that, while they are optimized to give the best trajectory based on the constraints of the basis polynomials, they cannot give the truly optimal solution described by the infinite dimensional optimal control problem, nor would an optimal solution be verifiable through Pontryagin's MP due to the lack of costate information. Also, maneuver duration is an issue that currently plagues both the direct and IDVD methods. For IDVD this is due to the additional coupling of the attitude and translational motion through the speed factor. If the speed factor is made small to accommodate constraints on translational motion and there is a nonzero requirement on a higher order derivative at an attitude maneuver endpoint (note this issue does not exist for translational maneuvers or rest-to-rest maneuvers), then the attitude maneuver may "wrap" or rotate beyond 2π in order to perform the maneuver to meet the endpoint requirements. This is because, for the given set higher order derivative in the time domain, a reduction of speed factor at the end point (which may be needed to accommodate maneuvers with excessive times) would lead to a larger value for the respective higher order derivative in the virtual time domain since λ appears in the denominator of the conversion multiplier from time to virtual domains. This also translates into a performing a larger maneuver in quaternion space to accommodate the higher derivative value in the virtual domain. While the maneuver would still be completely feasible, the extra rotation may be undesired in the final approach. Even though this is not an issue for the implementation discussed in this dissertation, it would need to be addressed for application on a subset of missions that have drastically different timescales associated with the attitude and translational maneuver. Although this is easily rectified by simply setting constraints on the speed factor so the extra rotation does not happen, it may hinder the resulting PI. Another approach is employing a trajectory

generation scheme for maneuvers starting farther away (far approach), which simply holds attitude fixed in the orbital frame (McCamish 2007) or pointing to the RSO, then switching to the final approach scheme discussed in this dissertation. This is in line with operational methodology that would want to be pointing at the target as long as possible in order to get the most up-to-date information on the tumbling RSO before performing the rapid final approach maneuver at the last possible instant. On the other hand, for the direct collocation methods, maneuvers that are particularly long in length are subject to large time gaps in between nodes, where no state or control information is available. Extra caution needs to be taken when employing such a solution.

Summarizing, two novel guidance strategies for final approach to a tumbling object are developed in this dissertation. One is based on Inverse Dynamics in the Virtual Domain, employing a newly formulated quaternion approach, and the other is a novel formulation that employs direct collocation methods for optimal trajectory generation. While the direct collocation method formulation is too computationally expensive to be employed, it provides a mathematically verifiable optimal baseline for the maneuvers. While the method based on IDVD cannot exactly match the optimal solution, it can provide a feasible that is optimized stemming from set of basis functions with a fraction of the computational time of the direct method. This and other attractive features discussed, allows for exploitation of several real time implementation concepts. It is recommended that the IDVD method be employed for situations where the computational cost of the solution outweighs the PI associated with the state and control histories. The direct collocation, or pseudospectral, method should be used in circumstances where computational time is not a concern, but a truly optimal solution based on the state and control variables is desired, such as base lining scenarios and finding limits of specific technologies.

D. POSSIBLE FUTURE DEVELOPMENTS

The following research problems remain open for possible follow-up efforts:

1. Investigate effects of different sets of basis function for trajectory generation and overall effect of using IDVD methods to seed initial guesses for pseudospectral optimal control solvers.
2. Analyze performance of trajectory tracking in simulation.
3. Experimental validation on real autonomous vehicles.
4. Integration with current research involving navigation and target identification.

LIST OF REFERENCES

- AFRL 2005: XSS-11, www.kirtland.af.mil/shared/media/document/AFD-070404-108.pdf [cited July 20, 2009].
- Bevilacqua, R., Romano, M., Yakimenko, O. (2009), 'Online Generation of Quasi-optimal Spacecraft Rendezvous Trajectories,' *Acta Astronautica*, 64 (2–3), pp. 345–358.
- Bilimoria, K. D., and Wie, B. (1993), 'Time-Optimal Three-Axis Reorientation of a Rigid Spacecraft,' *Journal of Guidance, Control, and Dynamics*, 16 (3), pp. 446–452.
- Bosse, A. B. (2004), 'SUMO: Spacecraft for the Universal Modification of Orbits,' *Proceedings of the SPIE Defense and Security Symposium*, pp. 36–46.
- Boyarko, G., Yakimenko, O., and Romano, M. (2009), 'Formulation and Analysis of Matching Points of Interest in Two-Spacecraft for Optimal Rendezvous,' *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, pp. 1–22.
- (2009b), 'Optimization of a Spacecraft Maneuver to Dock with a Tumbling Object,' *Proceedings of the AAS/AIAA Astrodynamics Specialist Conference*, pp. 1–18.
- Broad, W. (1992), 'Shuttle Fails Twice to Capture Satellite,' NY Times, Published: Monday, May 11, 1992.
- Bryson, A. E., Jr., and Ho, Y.-C. (1975), *Applied Optimal Control: Optimization, Estimation, and Control*, Taylor and Francis.
- Cowling, I., Yakimenko, O., Whidborne J., and Cooke, A. (2007), 'A Prototype of an Autonomous Controller for a Quadrotor UAV,' *European Control Conference*, pp. 1–8.
- Creamer, G. (2007), 'The SUMO/FREND Project: Technology Development for Autonomous Grapple of Geosynchronous Satellites,' *Advances in the Astronautical Sciences*, 128, pp. 895–910.
- Davis, T.M. (2003), 'XSS-10 Micro-Satellite Flight Demonstration Program,' *Proceedings of the 17th Annual AIAA/USU Conference on Small Satellites*, Logan, Utah, August 11–14.
- Davis, T., and Straight, S. (2006), 'Development of the Tactical Satellite 3 for Responsive Space Missions,' *Proceedings of the 4th Responsive Space Conference*, April 24–27, Los Angeles California.

- Fleming, A. (2004), 'Real time Optimal Slew Maneuver Design and Control,' unpublished Master's thesis, U.S. Naval Postgraduate School, Dept. of Mechanical and Aerospace Engineering.
- Gill, P. E., Murray, W., and Saunders, M. A. (1996), 'SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization,' Dept. of Operations Research, Stanford Univ., TR SOL 96-0, Stanford, California
- Greenwood, D. T. (1987), *Principles of Dynamics*, London: Prentice Hall.
- Henshaw, C. G. (2003), 'A Variational Technique for Spacecraft Trajectory Planning,' PhD Dissertation, Department of Aerospace Engineering, University of Maryland College Park.
- Huntington, G., and Rao, A. (2008a), 'Comparison of Global and Local Collocation Methods for Optimal Control,' *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 2, 2008, pp. 432–436.
- (2008b), 'Optimal Reconfiguration of Spacecraft Formations using Gauss Pseudospectral Method,' *AIAA Journal of Guidance Control and Dynamics*, Vol. 31, No. 3, pp. 689–698.
- Hurni, M. (2009), 'An information-centric approach to autonomous trajectory planning utilizing optimal control techniques,' Doctoral Dissertation, U.S. Naval Postgraduate School, Monterey, California.
- Jacobsen, S., Lee, C., Zhu, C., and Dubowsky, S. (2002), 'Planning of Safe Kinematic Trajectories for Free Flying Robots Approaching Uncontrolled Spinning Satellite,' *Proceedings of the ASME Design Engineering Technical Conferences and Computer and Information in Engineering Conference*, pp. 1–7.
- Jasiobedski, P., Greenspan, M., and Roth, G. (2001), 'Pose Determination and Tracking for Autonomous Satellite Capture,' *The 6th International Symposium on Artificial Intelligence, Robotics and Automation in Space*, pp. 1–8.
- Junkins, J., and Turner, J. (1986), *Optimal Spacecraft Rotational Maneuvers*, New York: Elsevier Science Ltd.
- Kennedy, F. (2008), 'Orbital Express: Accomplishments and Lessons Learned,' *Advances in the Astronautical Sciences*, 131, pp. 575–586.
- Kim, M., Kim, M. and Shin, S. (1995), 'A General Construction Scheme for Unit Quaternion Curves with Simple High Order Derivatives,' *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, pp. 369–376.

- Louembet, C., Cazaurang, F., Zolghardi, A., Charbonnel, C., and Pittet, C. (2007), 'Design of Algorithms for Satellite Slew Manoeuvre by Flatness and Collocation,' *Proceedings of the American Control Conference*, pp. 3168–3173.
- Ma, Z., Ma, O., Shashikanth, N. (2007), 'Optimal Approach to and Alignment with a Rotating Rigid Body for Capture,' *Journal of Astronautical Sciences*, 55 (4), 2007, pp. 407–409.
- Matsumoto, S., Dubowsky, S., Jacobsen, S., and Ohkami (2003a), Y., 'Approach Planning and Guidance for Uncontrolled Rotating Satellite Capture Considering Collision Avoidance,' *Proceedings of the 7th International Symposium on Artificial Intelligence, Robotics and Automation in Space: i-SAIRAS 2003*, pp. 1–8.
- (2003b), Y., 'Fly-By Approach and Guidance for Uncontrolled Rotating Satellite Capture,' *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, pp. 1–10.
- McCamish, S. (2007), Distributed Autonomous Control of Multiple Spacecraft During Proximity Operations, unpublished PhD dissertation, U.S. Naval Postgraduate School, Dept. of Engineering and Applied Sciences.
- McFarland, D., Swenson, E., Black, J., Cobb, R., and Fosbury, A. (2009), 'Near Real time Closed-Loop Optimal Control Feedback for Spacecraft Attitude Maneuvers,' *Proceedings of the AIAA Modeling and Simulation Technologies Conference*, pp. 1–19.
- McInnes, C. R. (1998), 'Satellite Attitude Slew Maneuver Using Inverse Control,' *The Aeronautical Journal*, 102, pp. 259–265.
- Milam, M. (2003), 'Real time Optimal Trajectory Generation for Constrained Dynamical Systems,' unpublished Doctoral thesis, California Institute of Technology, Dept. of Control and Dynamical Systems and Mechanical Engineering.
- NASA (2004), DART Mission, www.nasa.gov/missions/science/dart_into_space.html [cited July 20, 2009].
- Nolet, S (2007), 'Development of a Guidance, Navigation and Control Architecture and Validation Process Enabling Autonomous Docking to a Tumbling Satellite,' unpublished Doctoral thesis, Massachusetts Institute of Technology, Dept. of Aeronautics and Astronautics.
- , Kong, E., and Miller, D. W. (2005), 'Design of an Algorithm for Autonomous Docking with a Freely Tumbling Target,' in *Proceedings of Modeling, Simulation and Verification of Space-based Systems II*, 5799 (123), pp. 123–134.

- Pontryagin, L. S., Boltyanskii, V., Gamkrelidze, R., and Mishchenko, E., *The Mathematical Theory of Optimal Processes*, Pergamon, New York, 1964.
- Rao, A. V., Benson, D. A., Darby, C. L., Patterson, M. A., Francolin, C., Sanders, I., and Huntington, G. T. (2008), 'GPOPS: A MATLAB Software for Solving Multiple-Phase Optimal Control Problems Using the Gauss Pseudospectral Method,' *ACM Transactions on Mathematical Software*.
- Ross, I. M. (2004), *User's Manual for DIDO: A MATLAB Application Package for Solving Optimal Control Problems*, Tomlab Optimization, Sweden.
- Romano, M., Friedman, A., and Shay, T. (2007), 'Laboratory Experimentation of Autonomous Spacecraft Approach and Docking to a Collaborative Target,' *Journal of Spacecraft and Rockets*, 44 (1), pp. 164–173.
- , and Hall, J. (2006), 'A Testbed for Proximity Navigation and Control of Spacecraft for On-orbit Assembly and Reconfiguration,' in *Proceedings of the AIAA Space 2006 Conference and Exhibit*, pp. 1–11.
- Sakawa, Y. (1999), 'Trajectory Planning of a Free-Flying Robot by Using the Optimal Control,' *Optimal Control Applications and Methods*, 20 (5), pp. 235–248.
- Sarkka, S. (2007), 'Notes on quaternions,' Department of Biomedical Engineering and Computational Science (BECS) Centre of Excellence in Computational Complex Systems Research Technical Report.
- Sidi, M. (1997), *Spacecraft Dynamics and Control*, New York, NY, Cambridge University Press.
- Singh, G., and Hadaegh, F. Y. (2001), 'Collision Avoidance Guidance for Formation-Flying Applications,' *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, pp. 1–11.
- Tasker, F., and Henshaw, C. (2008), 'Managing Contact Dynamics for Orbital Robotic Servicing,' *AIAA SPACE 2008 Conference & Exposition*.
- Titterton, D., and Weston, J. (1997), *Strapdown Inertial Navigation Technology*, United Kingdom: Peter Pregrinus Ltd.
- Tsuda, Y., and Nakasuka, S. (2003). 'New Attitude Motion Following Control Algorithm for Capturing Tumbling Object in Space,' *Acta Astronautica*, 53 (11), pp. 847–861.
- Vadali, S. R., and Junkins, J. L. (1984), 'Optimal Open-Loop and Stable Feedback Control of Rigid Spacecraft Attitude Maneuvers,' *The Journal of the Astronautical Sciences*, 32 (2), pp. 105–122.

- Vaidyanathan, R., Quinn, R., Ritzmann R., and Prince T. (2001), ‘An Insect-Inspired Endgame Targeting Reflex for Autonomous Munitions,’ *International Conference on Intelligent Robots and Systems*, pp. 1–8.
- Vallado, D. A. (2001), *Fundamentals of Astrodynamics and Applications*, El Segundo, CA: Microcosm, Inc.
- Yakimenko, O., Xu, Y., and Basset, G. (2008), ‘Computing Short-Time Aircraft Maneuvers Using Direct Methods,’ *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, pp. 1–23.
- Yakimenko, O. (2000), ‘Direct Method for Rapid Prototyping of Near Optimal Aircraft Trajectories,’ *AIAA Journal of Guidance, Control, and Dynamics*, 23(5), pp. 865–875.
- , and Slegers, N. (2009), ‘Using Direct Methods for Terminal Guidance of Autonomous Aerial Delivery Systems,’ *Proceedings of the European Control Conference*, pp. 2372–2377.
- Yan, H., Fahroo, F., Ross, I. M. (2002), ‘Real time Computation of Neighboring Optimal Control Laws,’ *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, pp. 1–7.
- Wie, B. (1998), *Space Vehicle Dynamics and Control*, Reston, VA: American Institute of Aeronautics and Astronautics, Inc.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX. QUATERNION PROPERTIES

A quaternion is a four-dimensional vector that is used to express orientation of a rigid body. It is composed of a scalar, s , and a vector portion, \mathbf{v} . There are several representations and formulation of quaternions and their associated properties in literature. Throughout this dissertation, the properties in this Appendix will be employed (Titterton and Weston 1997; Sarkka 2007).

Quaternion Expression:

$$\mathbf{q} = \begin{pmatrix} s \\ \mathbf{v} \end{pmatrix} = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{pmatrix} \quad (151)$$

Quaternion Length:

$$|\mathbf{q}| = \sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2} \quad (152)$$

Quaternion Logarithm:

$$\ln(\mathbf{q}) = \begin{pmatrix} \ln(|\mathbf{q}|) \\ \frac{\mathbf{v}}{|\mathbf{v}|} \arccos\left(\frac{s}{|\mathbf{q}|}\right) \end{pmatrix} \quad (153)$$

Quaternion Exponential:

$$\exp(\mathbf{q}) = \exp(s) \begin{pmatrix} \cos(|\mathbf{v}|) \\ \frac{\mathbf{v}}{|\mathbf{v}|} \sin(|\mathbf{v}|) \end{pmatrix} \quad (154)$$

Quaternion Conjugate:

$$\mathbf{q}^* = \begin{pmatrix} q_1 \\ -q_2 \\ -q_3 \\ -q_4 \end{pmatrix} \quad (155)$$

Quaternion Inverse:

$$\mathbf{q} = \frac{\mathbf{q}^*}{|\mathbf{q}|^2} \quad (156)$$

Quaternion Product:

$$\mathbf{q} = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{pmatrix} \quad \mathbf{p} = \begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{pmatrix} \quad (157)$$

$$\mathbf{q}^* \mathbf{p} = \begin{pmatrix} q_1 p_1 - q_2 p_2 - q_3 p_3 - q_4 p_4 \\ q_2 p_1 + q_1 p_2 - q_4 p_3 + q_3 p_4 \\ q_3 p_1 + q_4 p_2 + q_1 p_3 - q_2 p_4 \\ q_4 p_1 - q_3 p_2 + q_2 p_3 + q_1 p_4 \end{pmatrix} \quad (158)$$

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California
3. Marcello Romano
Naval Postgraduate School
Monterey, California
4. Oleg Yakimenko
Naval Postgraduate School
Monterey, California
5. Brij Agrawal
Naval Postgraduate School
Monterey, California
6. Xiaoping Yun
Naval Postgraduate School
Monterey, California
7. Frank Chavez
Air Force Research Laboratory
Albuquerque, New Mexico
8. William K. Schum
Air Force Research Laboratory
Albuquerque, New Mexico
9. Rich de Jonckheere
Air Force Research Laboratory
Albuquerque, New Mexico
10. Jim Riker
Air Force Research Laboratory
Albuquerque, New Mexico
11. Wade Vaught
Air Force Research Laboratory
Albuquerque, New Mexico